OPTOOPCSERVER™ USER'S GUIDE

Form 1439-230725—July 2023

OPTO 22
Your Edge in Automation.™

43044 Business Park Drive • Temecula • CA 92590-3614
Phone: 800-321-OPTO (6786) or 951-695-3000
Fax: 800-832-OPTO (6786) or 951-695-2712
www.opto22.com

Product Support Services
800-TEK-OPTO (835-6786) or 951-695-3080
Fax: 951-695-3017
Email: support@opto22.com
Web: support.opto22.com

OptoOPCServer User's Guide
Form 1439-230725—July 2023

The information in this manual has been checked carefully and is believed to be accurate; however, Opto 22 assumes no responsibility for possible inaccuracies or omissions. Specifications are subject to change without notice.

Opto 22 warrants all of its products to be free from defects in material or workmanship for 30 months from the manufacturing date code. This warranty is limited to the original cost of the unit only and does not cover installation, labor, or any other contingent costs. Opto 22 I/O modules and solid-state relays with date codes of 1/96 or newer are guaranteed for life. This lifetime warranty excludes reed relay modules, *groov* and SNAP serial communication modules, SNAP PID modules, and modules that contain mechanical contacts or switches. Opto 22 does not warrant any product, components, or parts not manufactured by Opto 22; for these items, the warranty from the original manufacturer applies. Refer to Opto 22 form 1042 for complete warranty information.

Opto 22
Your Edge in Automation.

Table of Contents

1: Introduction

OptoOPCServer™ is an OPC-DA 2.0-compliant server used to communicate with these Ethernet-based Opto 22 devices:

- ***groov* EPIC processors**
- ***groov* RIO units**
- **SNAP industrial controllers**, including the standalone SNAP PAC S-series, the rack-mounted SNAP PAC R-series, the SNAP-LCE, and the SNAP-LCM4 with an Ethernet adapter card
- **SNAP brains** in the SNAP PAC EB-series, SNAP Ultimate I/O™, SNAP Ethernet I/O™, and SNAP Simple I/O™ families
- **E1 and E2 brain boards**

This guide assumes that you already have a basic understanding of OPC. If you need to learn about OPC, a good place to start is the OPC Foundation, at https:/opcfoundation.org.

ABOUT THIS GUIDE

This user's guide shows you how to set up and use OptoOPCServer.

This guide assumes that you are already familiar with Microsoft® Windows® on your personal computer. In addition, a basic understanding of network security is helpful.

Here's what is in this user's guide:

Chapter 1: Introduction—About the guide, system requirements, and how to reach Opto 22 Product Support.

Chapter 2: Installing and Using OptoOPCServer—System architecture and how to set up OptoOPCServer.

Chapter 3: Configuring DCOM in Windows—How to configure Windows to allow communication between OptoOPCServer and OPC clients on different computers.

Chapter 4: Opto 22 OPC Item Definitions—Syntax and item IDs used by OptoOPCServer messages, including I/O point data, memory map information, and PAC Control strategy variables.

Chapter 5: Debugging and Troubleshooting—Debugging in the OptoOPCServer Window, using the registry checker, testing a browser interface, and an explanation of data quality messages.

Appendix A: Using Prosys Client with OptoOPCServer—How to obtain, install, and configure the Prosys OPC Test Client, which can be used to test OptoOPCServer and the browser databases you create.

RELATED DOCUMENTATION

See the following documents for additional information:

For this information	See this guide	Form #
Designing flowchart-based control programs for *groov* EPIC and SNAP PAC controllers	PAC Control User's Guide	1700
	PAC Control Command Reference	1701
Installing and using the *groov* EPIC processor	*groov* EPIC User's Guide	2267
Installing and using a *groov* RIO unit	*groov* RIO User's Guide	2324
Configuring SNAP I/O points and system functions	PAC Manager User's Guide	1440
Installing and using Opto 22's SNAP PAC R-series programmable automation controllers	SNAP PAC R-Series Controllers User's Guide	1595
Installing and using Opto 22's SNAP PAC S-series programmable automation controllers	SNAP PAC S-series Controllers User's Guide	1592
Installing and using SNAP PAC Brains	SNAP PAC Brains User's Guide	1690
Installing and using E1 and E2 brain boards	E1 and E2 User's Guide	1563

FOR HELP

If you have problems installing or using OptoOPCServer, first check this guide and the Troubleshooting section of the user's guide for your Opto 22 hardware. If you cannot find the help you need in the guides or on the Opto 22 Web site, contact Opto 22 Product Support.

Phone: 800-TEK-OPTO
(800-835-6786 toll-free in the U.S. and Canada)
951-695-3080
Monday through Friday,
7 a.m. to 5 p.m. Pacific Time

Email: support@opto22.com

Opto 22 website: www.opto22.com

When calling for technical support, you can help us help you *faster* if you provide the following information to the Product Support engineer:

- A screen capture of the Help > About dialog box showing software product and version (available by clicking Help > About in the application's menu bar).
- Opto 22 hardware part numbers or models that you're working with.
- Firmware version:
  - For *groov* EPIC processors and *groov* RIO modules: available in *groov* Manage by clicking Info and Help > About.
  - For SNAP controllers and brains: available in PAC Manager by clicking Tools > Inspect.
- Specific error messages you received.
- Version of your computer's operating system.
- For PAC Control, PAC Display, OptoOPCServer, or PAC Manager, you may be requested to provide additional information, such as log or dump files. You can find these files in a support files sub-folder:
  a. On your Windows Desktop, double-click the PAC Project 10.5 folder.
  b. Double-click Support Files.
  c. Double-click on the appropriate shortcut to open the sub-folder containing the requested files.

  Note: *PAC Control, PAC Display, OptoOPCServer, and PAC Manager create appropriate sub-folders when they create diagnostic log or dump files. If they have not created these files, the sub-folder may not exist; in this case, the shortcut will not work.*

System Requirements

Here's what you need to install and run OptoOPCServer:

- A computer with at least the minimum processor and memory required for your version of Microsoft® Windows®, and Ethernet capability. Additional memory and a better CPU may be required for some configurations. We recommend staying away from low-end CPUs and using one of the following operating systems:
  - Microsoft® Windows® 10 Professional (32-bit or 64-bit) or Windows 11 Professional
  - (OptoOPCServer and OptoDataLink) Windows Server® 2012 R2, 2016, 2019, and 2022; however, you must install .NET Framework 4.0 prior to installing PAC Project on these operating systems

  NOTE: PAC Project cannot be installed on Windows XP or older Windows operating systems. Other versions of Windows operating systems and embedded Windows operating systems are not supported.

- Ethernet capability
- VGA or higher resolution monitor. Minimum size: 800x600 with small fonts
- Mouse or other pointing device
- (Optional) Installed Windows printer
- At least 17.5 MB of available hard drive space (for OptoOPCServer only) or up to 337.1 MB if you are installing PAC Project Professional and its default components.
- If your PAC Display™ Professional project accesses an M4-series controller (such as a SNAP-LCM4 or M4RTU) via an Ethernet connection, controller firmware version R4.1a or newer is required. In addition, in order to access strings or string tables, controller firmware R4.1d or newer is required.

Installation Note for Windows

Windows requires that you configure OPC client and OPC server identities and privileges to allow communication when the OPC Server and OPC client are on different computers. Chapter 3: Configuring DCOM in Windows describes this configuration.

2: Installing and Using OptoOPCServer

This chapter describes system architecture, how OptoOPCServer works, and how to install and use OptoOPCServer. This chapter includes the following topics:

SYSTEM ARCHITECTURE

The OptoOPCServer uses OPC protocol to communicate with OPC clients, and an Opto 22 proprietary protocol to communicate with Opto 22 devices. OptoOPCServer supports:

- *groov* EPIC processors (PAC Control strategy tags)
- *groov* RIO units (PAC Control strategy tags)
- Ethernet-based I/O units in the SNAP PAC, Ultimate I/O, Ethernet I/O, and Simple I/O product families
- SNAP PAC and Ultimate I/O controller families (PAC Control and ioControl strategy tags)

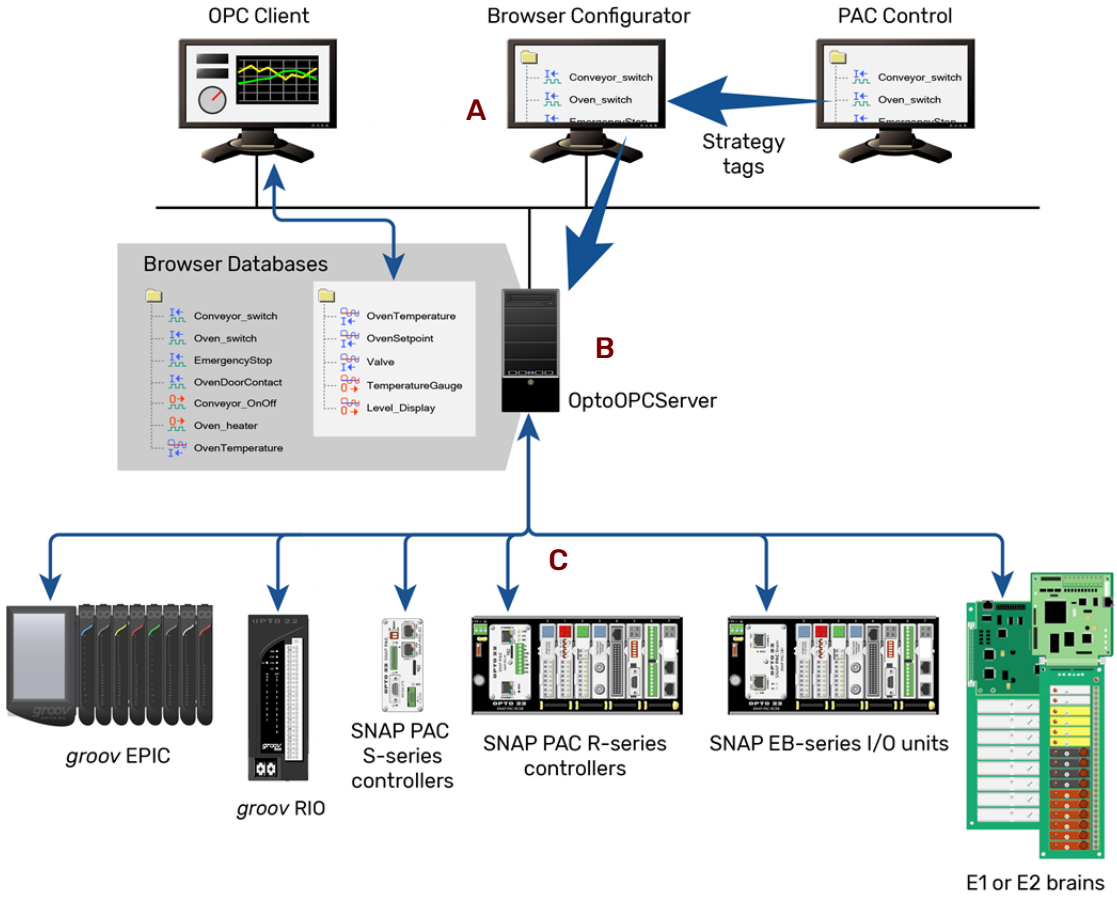
NOTE: While a FactoryFloor controller with a M4SENET-100 Ethernet communications card installed is supported, it is not recommended due to throughput issues.

Before OptoOPCServer can communicate with Opto 22 controllers, a few steps are required:

- First, commission your Opto 22 device.
  - For *groov* EPIC processors, use *groov* Manage and assign its IP address. If the processor has dual Ethernet interfaces, you can assign a secondary IP address. You'll want to make sure you also configure the firewall properties to allow communication through the secondary IP address. For details, see the *groov* EPIC User's Guide (form 2267).
  - For *groov* RIO units, use *groov* Manage and assign its IP address. For details, see the *groov* RIO User's Guide (form 2324).
  - For SNAP PAC devices, use PAC Manager software and assign its IP address. If your device has dual Ethernet interfaces, you can also use PAC Manager to assign a secondary IP address. For details, see the PAC Manager User's Guide (form 1704).
- Next, use Opto 22 PAC Control software to create and configure your control strategy (a flowchart-based control program that defines workflows). Within a strategy, select the processor, controller, and I/O units that perform the work.
- Finally, download the strategy to the controller and run the strategy. OptoOPCServer uses the IP addresses configured within the strategy to communicate with the devices.

For information on how to create and configure strategies, see the *PAC Control User's Guide* (form 1700).

The image below shows a sample system using OptoOPCServer.



A The Opto Browser Configurator (installed as part of OptoOPCServer) allows you to browse to other computers on the network. OPC clients connect to browser databases through OptoOPCServer.
B Opto Browser Configurator loads browser database files to a local or remote OptoOPCServer.
C OptoOPCServer scans devices for tags chosen by the OPC clients.

Ethernet Link Redundancy

To protect your control system against physical link failure, you can set up Ethernet link redundancy in SNAP PAC controllers and R-series I/O units.

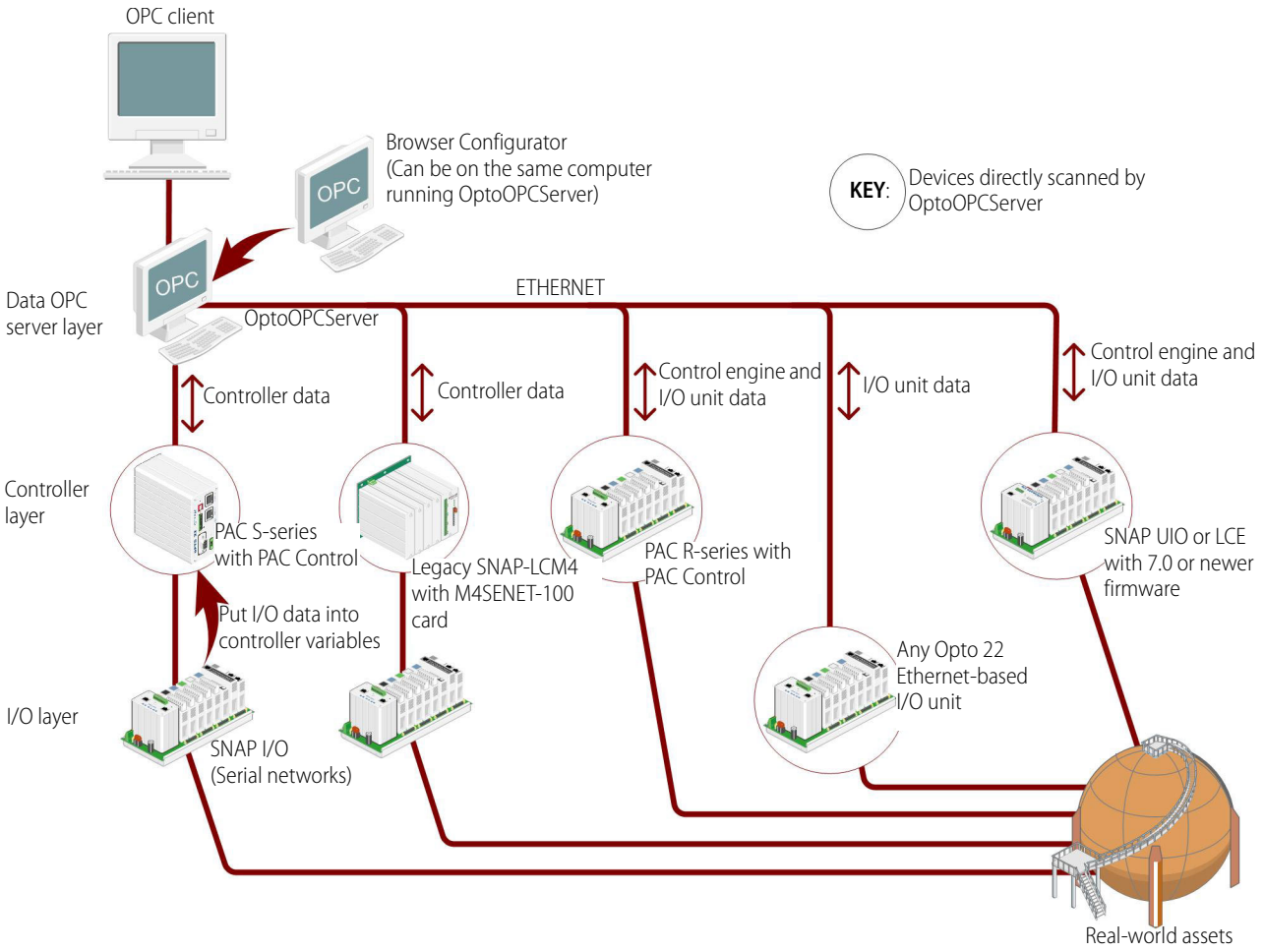
In PAC Manager, you assign a primary IP address to the device's Ethernet 1 interface (Tools > Assign IP Address). Then, you can assign a secondary IP address to the Ethernet 2 interface (Tools > Inspect > Status Write button | ETHERNET 2 Interface).

After the device has been commissioned in PAC Manager, you configure a PAC Control Professional strategy that uses the controller's Ethernet interfaces for redundant networking.

When a PAC Display or OptoDataLink project uses a strategy with redundant controllers (and, optionally, redundant I/O units), it provides both the primary and secondary IP addresses to the OptoOPCServer scanner. If a communication failure occurs between OptoOPCServer and the primary IP address, communication switches[1] to the secondary address.

For more information on Ethernet link redundancy, see:

- "Configuring Devices," in the PAC Manager User's Guide (form 1704)
- "Using Ethernet Link Redundancy in PAC Control," in the PAC Control User's Guide (form 1700).



Accessing Legacy Controllers

OptoOPCServer can access any ioControl (Ultimate or LCE) controller via Ethernet as long as each controller has version R7.0 or newer firmware.

OptoOPCServer can access any M4-series OptoControl controller via Ethernet as long as the controller has the optional M4SENET-100 card installed and controller firmware version R4.1a or newer. To access strings or string tables, controller firmware R4.1d or newer is required.

INSTALLING OPTOOPCSERVER

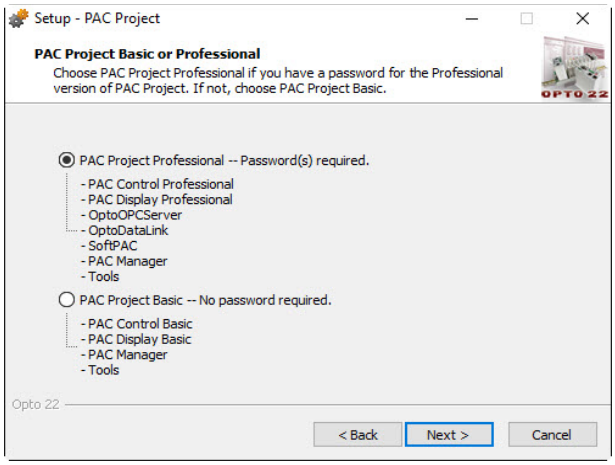
NOTE: If you are using older, legacy controllers or brains with OptoOPCServer, see the SNAP PAC System Migration Technical Note (form 1688) for important information.

[1] Switching is automatic when Ethernet link redundancy is configured in PAC Display or in OptoDataLink. To enable automatic switching when using third-party OPC client software, see "Redundancy with Third-Party OPC Clients" on page 60.

To open OptoOPCServer:

OptoOPCServer is part of the PAC Project™ Professional Software Suite, which you can download from our Opto 22 website at www.opto22.com. When you install OptoOPCServer, both the server components and the client components (including Browser Configurator) are installed.

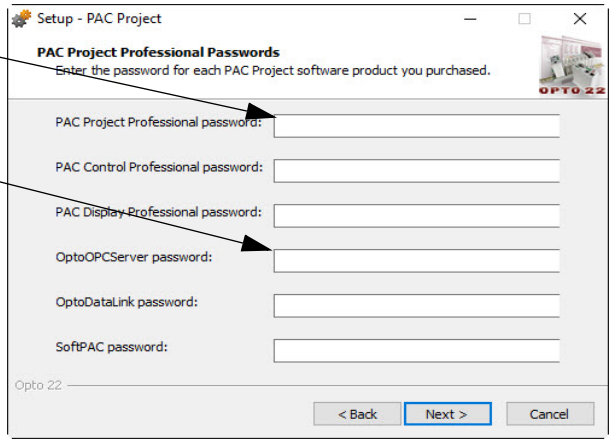
1. When you have a password (provided by your Opto 22 Inside Sales representative), download and run the PAC Project installer.
2. When prompted by the installer, select PAC Project Professional.



3. If you purchased the PAC Project Pro Software Suite, type the password in the PAC Project Professional password field.

   If you purchased only OptoOPCServer, type the password in the OptoOPCServer password field.

Type your PAC Project Professional password here
or
type your OptoOPCServer password here.



To start OptoOPCServer**:**

- In Windows, click the Windows Start button, and then click Opto 22 > OptoOPCServer 10.5.

To start Opto Browser Configurator:

- In Windows, click the Windows Start button, and then click Opto 22 > Browser Configurator 10.5.

NOTE: If you plan to use the OPC server and client components separately on different computers, additional Windows configuration is required on both computers. For more information, see Chapter , "3: Configuring DCOM in Windows."

NOTE: If Windows reports an error reading or finding a DLL when you try to start OptoOPCServer or Browser Configurator, see "Problems Due to Microsoft Fast Startup/Fast Reboot" on page 63.

If you have trouble installing OptoOPCServer, contact Opto 22 Product Support at 800-835-6786 (toll-free in the U.S. and Canada) or 951-695-3080.

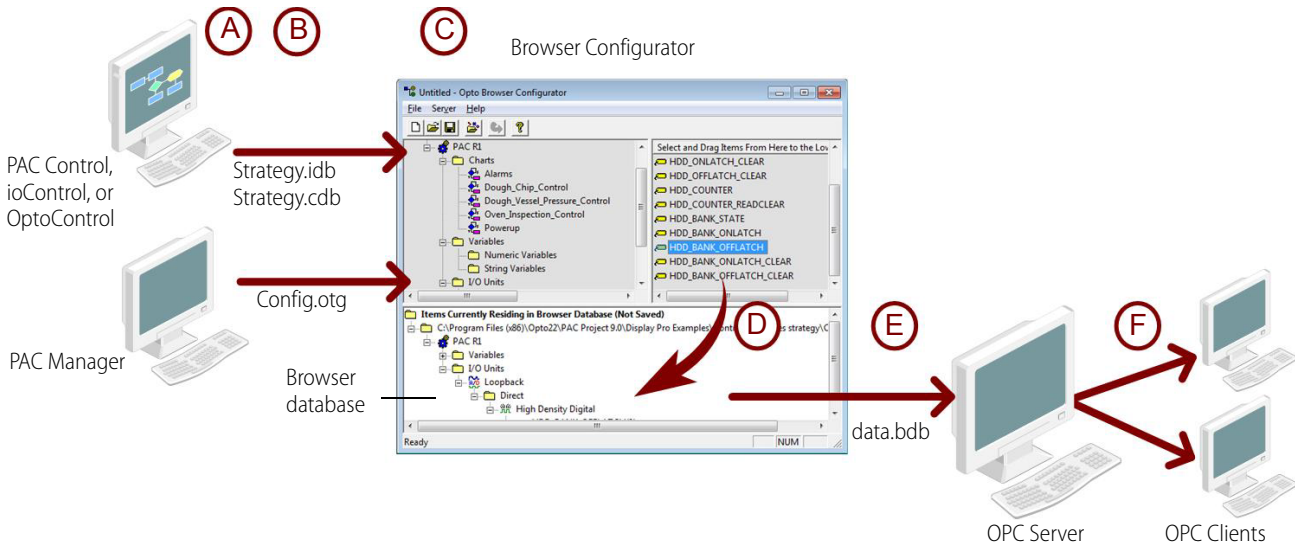
SETTING UP OPTOOPCSERVER

If your OPC client supports the OPC 2.0 tag browser interface, the PAC Control, ioControl, or OptoControl tags needed for communication with OPC clients can be automatically copied from the strategy and I/O configuration files. In OptoOPCServer, these files are called *browser items*.

Browser items from multiple controllers and brains can be opened in the Opto Browser Configurator and some or all of their tags copied to a single file referred to as a *browser database*. A browser database is loaded to OptoOPCServer to expose I/O and controller tags to OPC clients. You can create one or several browser databases as needed.

For initial setup, you will use the Opto Browser Configurator. After setup, OptoOPCServer starts automatically when an OPC client accesses it and runs without a visible window. For debugging the OPC server, you can open OptoOPCServer before an OPC client is connected to it, launch the client, and then observe OptoOPCServer to see the information requested by the client. (See Chapter 4: Opto 22 OPC Item Definitions for more information.)

You'll use the following steps to set up OptoOPCServer:



A. Configure Windows for OPC Clients on Other Computers

If there are OPC clients on computers other than where OptoOPCServer resides, in order to enable OPC communications between the computers you must complete additional Windows configuration steps on computers running the OPC client and the computer running the OptoOPCServer. For steps, see Chapter 3: Configuring DCOM in Windows.

B. Create Browser Items

You create browser items when you create a strategy in PAC Control or create I/O unit configuration files in PAC Manager. Note that you cannot configure *groov* EPIC and *groov* RIO devices with PAC Manager; therefore, you can't create I/O unit configuration files for them. To create browser items for these devices, you create a strategy in PAC Control. For more information on creating strategies and I/O unit configuration files, see:

- *PAC Control User's Guide* (form 1700)
- *PAC Manager User's Guide* (form 1704)

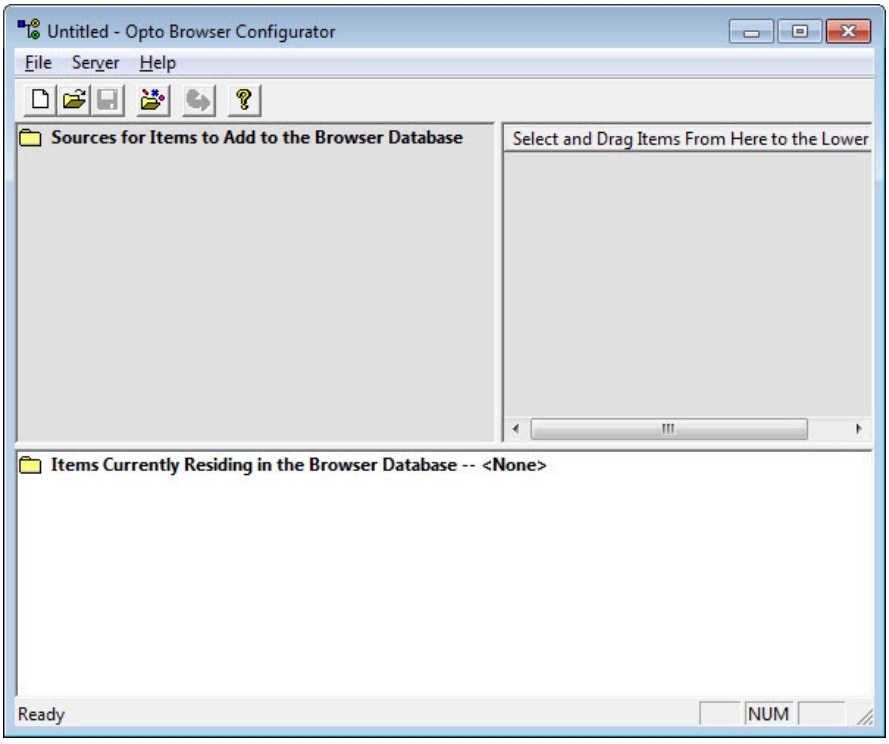
C. Open Browser Items File in Opto Browser Configurator

Browser items consist of:

- Tags from I/O unit configuration files created with PAC Manager (or ioManager)
- Strategy tags that reside in PAC Control (or ioControl or OptoControl) strategies

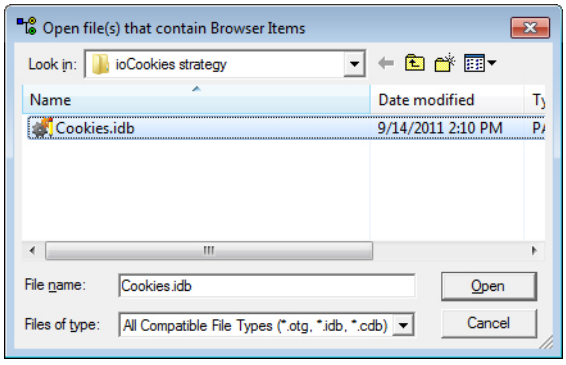
Items from any of these types of files can be added to an Opto Browser Configurator browser database. A browser database file can contain a mixture of items from single or multiple strategies, or I/O Unit configurations.

To open a browser items file in Opto Browser Configurator, you need to know the location of the browser item you created in PAC Manager (*.otg), PAC Control (*.idb), ioControl (*.idb), or OptoControl (*.cdb).The Opto Browser Configurator window appears with a new, untitled browser database.



1. From the File menu, choose Browser Items > Open Browser Items File.

2. In the Open File(s) dialog box, navigate to the .otg, .idb, or .cdb file you want to use as a source for OPC items.

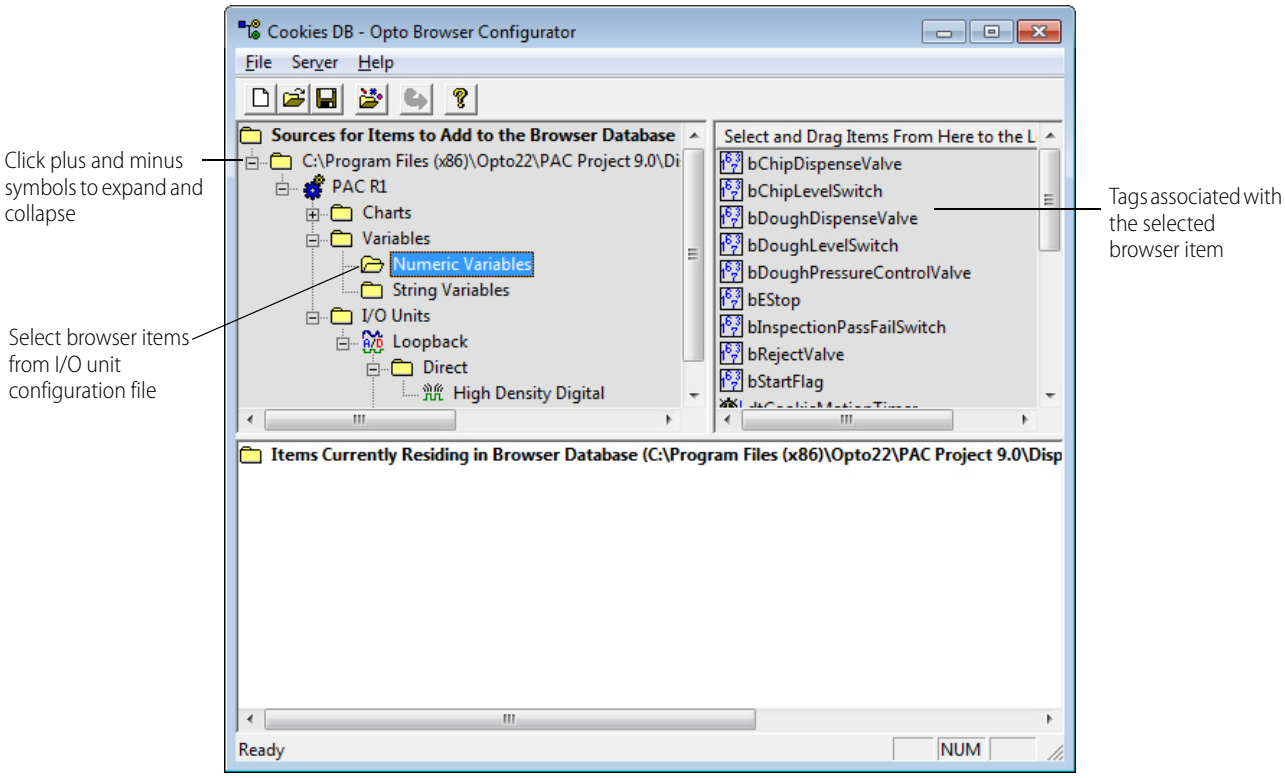


3. Highlight the filename and click Open.

Depending on the type of file you opened, see the appropriate section below.

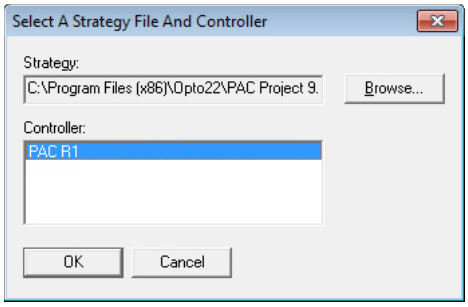
For PAC Manager or ioManager (.otg) Files

If you opened an .otg file for an I/O unit, the contents are now shown in the left pane of the Opto Browser Configurator. The right pane displays the specific tags associated with the browser item selected in the left pane.



For PAC Control, ioControl, OptoControl Strategy (.idb or .cdb) Files

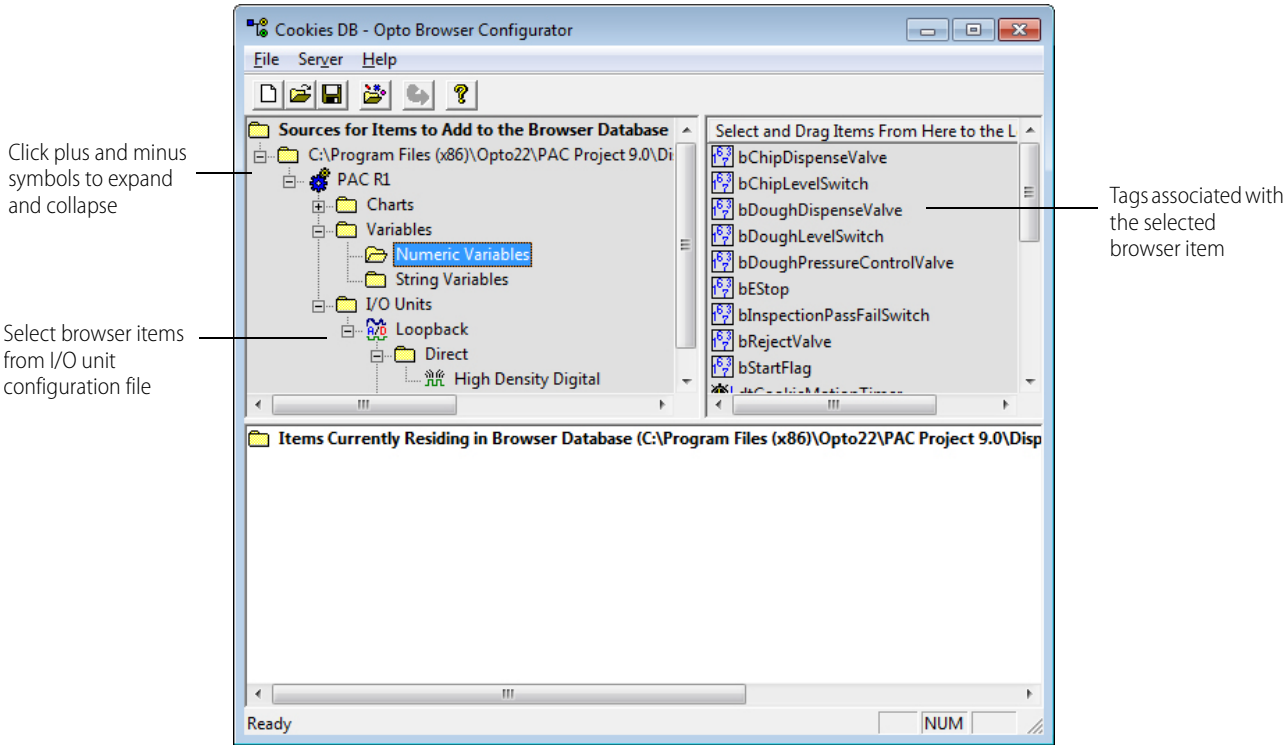
If you opened an .idb or .cdb file, the following dialog box appears.



NOTE: If no controller is listed, the strategy may not have a valid controller. To correct this, open the strategy (in PAC Control, ioControl, or OptoControl), configure a controller, and then save the strategy.

4. In the "Select a Strategy File and Controller" dialog box, click a controller to select it.
   To select multiple controllers, hold down Shift while clicking the right mouse button.
5. Click OK.
6. To select tags from additional strategies or .otg files, repeat steps 1 and 2.

The browser items are now shown in the left pane of the Opto Browser Configurator. The right pane displays the specific tags associated with the browser item selected in the left pane.



D. Build Browser Database

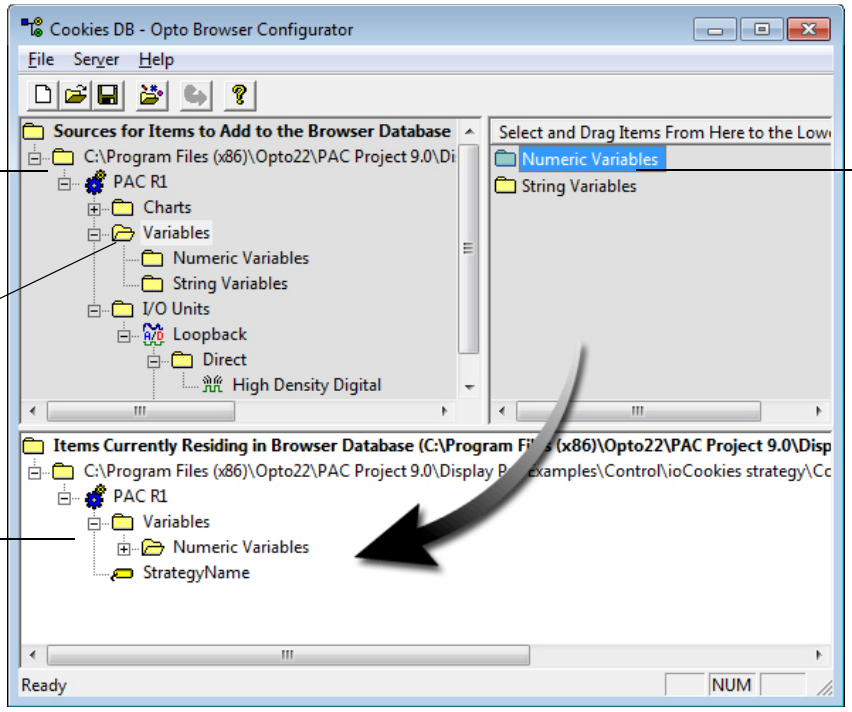
NOTE: High Density Digital tags require special treatment. See "Defining High-Density Digital Tags" on page 17.

After opening one or more browser items files, you can add tags to your browser database. The source files of any tag in your browser database remain referenced, so you will not need to add the same browser items again if you reopen your browser database. However, if your source file has new tags and you want to expose them to OPC clients, you will need to reopen the file to get the new tags.

To build the browser database, do the following:

1. Use the hierarchical tree in the top left side of the application window to locate specific tags.
   Click the plus symbols to expand groups. Click the minus symbols to collapse groups.

2. Select an item or folder in the top left source pane to display all of the selection's associated tags in the top right pane.



3. Select the tag in the top right pane and drag it to the bottom pane. You can select single or multiple tags.

   NOTE: For high-density digital tags, you need to provide additional information. See "Defining High-Density Digital Tags" on page 17.

   This technique works at all layers of the hierarchy. For example, if you want to copy all of the numeric variables of a strategy into the browser database, select the Variables folder and then drag the Numeric Variables folder from the right pane into the browser database.

   To select tags in a table:

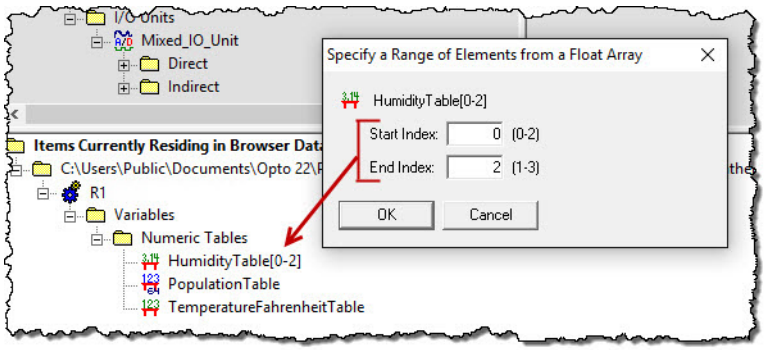
   a. Select a table in the top right pane and drag it to the bottom pane.

   b. In the lower pane, right-click the table to display the pop-up menu.

   To select a *range* of table elements:

   – In the pop-up menu, select Specify Array Range of Elements.

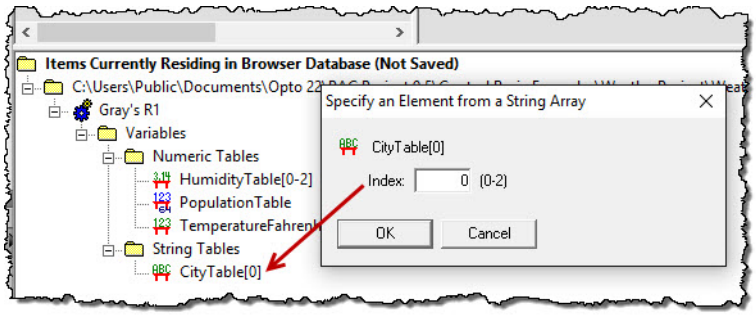
– In the "Specify a Range of Elements" dialog box, type the start index number in the Start Index text box, and the end index number in the End Index text box. Then click OK.
The range is displayed in brackets next to the table name.



To select a *single* table element:

– In the pop-up menu, select Specify Array Element.

– In the "Specify an Element" dialog box, type the index number. Then click OK.
The element number is displayed in brackets next to the table name.

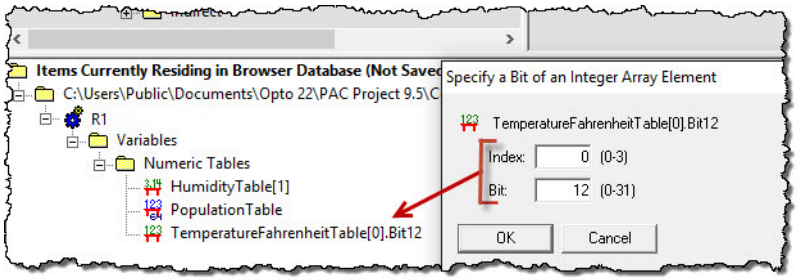


To select bits in an integer table:

c. Select the table in the top right pane and drag it to the bottom pane.

d. In the lower pane, right-click the table and select "Specify Integer Array Bit of Element" from the pop-up menu.

e. In the "Specify a Bit of an Integer Array Element" dialog box, type the index number in the Index text box, and the bit number in the Bit text box. Then click OK.
The index and bit are displayed in brackets [ ] next to the table name.

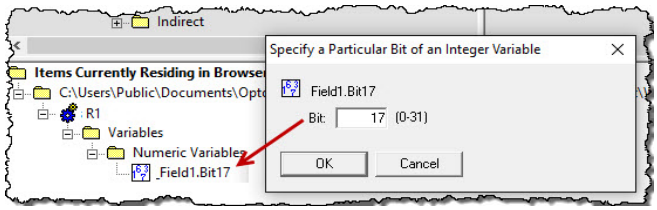


To select a bit in an integer variable:

f. Select the integer variable in the top right pane and drag it to the bottom pane.

g. In the lower pane, right-click the integer variable and select "Specify Integer Bit" from the pop-up menu.

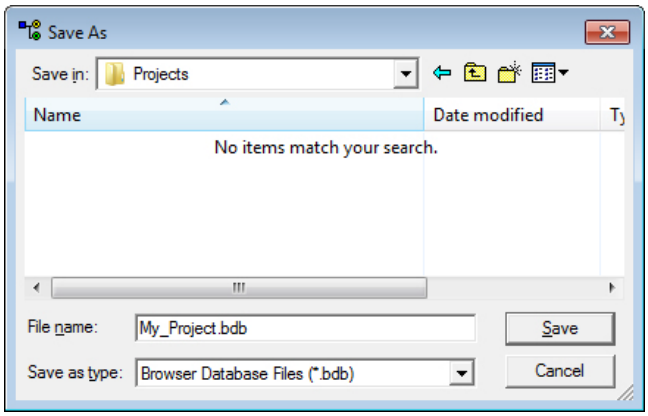
h. In the "Specify a Particular Bit" dialog box, type the bit number in the Bit text box. Then click OK. The bit number is displayed next to the variable name.



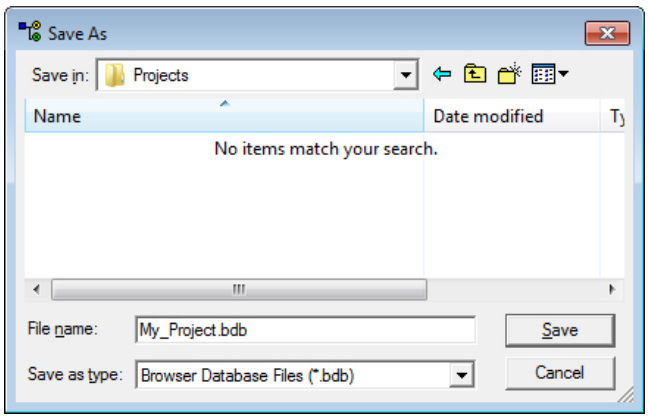
4. Repeat Steps 2 and 3 until all the desired tags are in the browser database.
5. Verify that all desired tags are in the browser database.

   The bottom pane displays the browser database. Expand items to check that you've included all the tags you intended to include. If you find tags you didn't intend to include, you can select these and press the Delete key to remove them.
6. To save your browser database, select Save from the File menu.
7. In the Save As dialog box, enter a file name and then click Save.

   If you will be uploading multiple browser database files, make sure each file has a unique name. If you load a browser database with the same name as a previous one, the previous database will be overwritten.



8. In the next dialog box, type a meaningful description, and then click OK.



A meaningful description is especially helpful if you plan to load multiple browser databases to your OptoOPCServer.

The saved browser database must be loaded to the OPC server before it is available to OPC clients. See the next section, "E. Load Browser Database to OptoOPCServer."

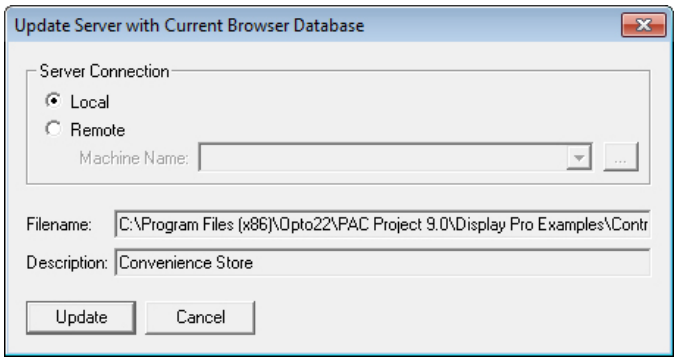
E. Load Browser Database to OptoOPCServer

You use the Opto Browser Configurator to load browser databases to OptoOPCServer. Once loaded to the OPC server, OPC client programs will have access to the tags through their browser capability.

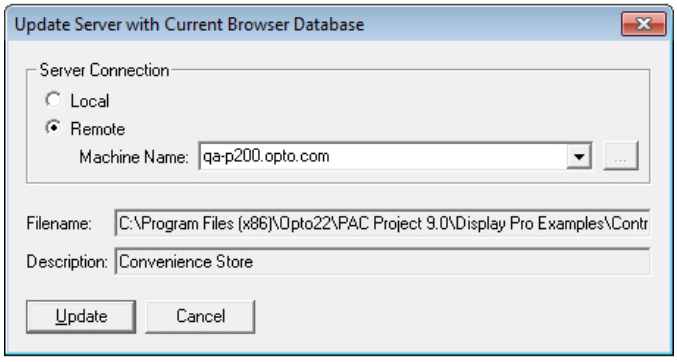
Loading a browser database overwrites a previously loaded database of the same name. Use unique filenames to upload multiple browser database files.

To upload a browser database, do the following:

1. If you do not have a browser database open, choose File > Open and navigate to your browser database.
2. Choose Server > Update Server with Current Browser Database.
   The Update Server dialog box appears.



3. If OptoOPCServer resides on the same workstation as Opto Browser Configurator, choose Local. However, if OptoOPCServer resides on another computer on the network, choose Remote.
   If you chose Remote, type the name of the computer running OptoOPCServer.



   The computer name is the network identification of the computer running OptoOPCServer. You can find its computer name by viewing the computer's System Properties.
4. Click the Update button.
   If you are overwriting an existing browser database of the same name, you'll see a warning. Click Yes to proceed.
5. Click OK to acknowledge the update.

F. Connect OPC Client to OptoOPCServer

Connect your OPC client to OptoOPCServer. When configuring your OPC client you will need to select the server named "Opto22.OpcServer.2."

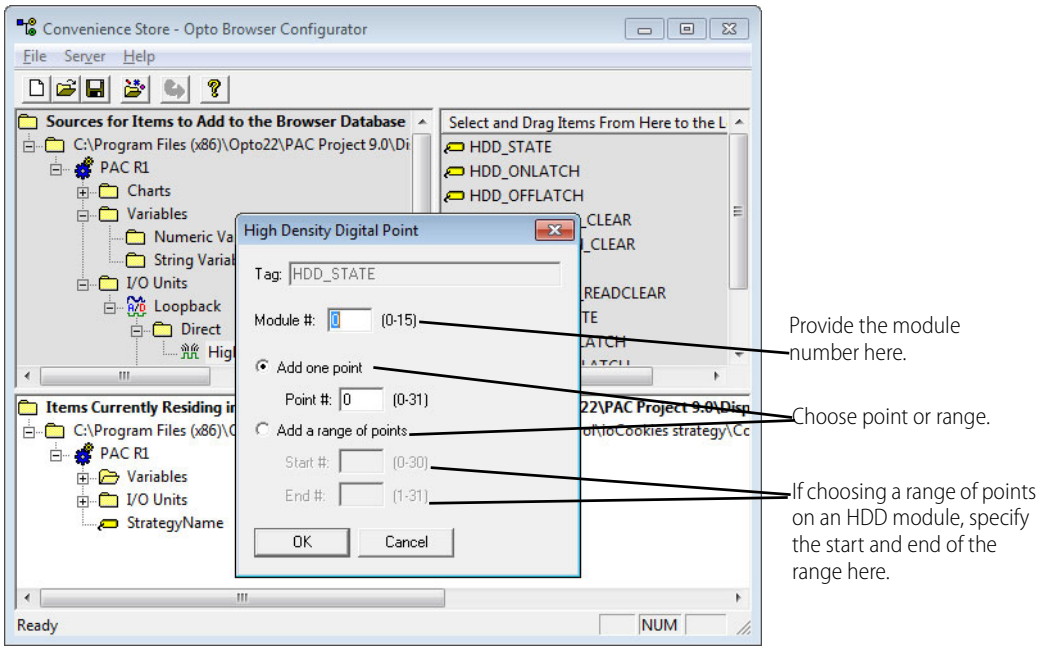
Prosys OPC Client is a free stand-alone OPC client application that you can use to test OptoOPCServer and the browser databases you create. This application is available as a free download from the Opto 22 FTP site. For instructions to obtain, install, and configure Prosys OPC Client, see "Appendix A: Using Prosys Client with OptoOPCServer."

DEFINING HIGH-DENSITY DIGITAL TAGS

Tags for points on SNAP high-density digital modules are handled somewhat differently from other tags in OptoOPCServer.

1. Do not drag the group High Density Digital into the browser database. Instead, select High Density Digital to display the specific tags, such as HDD_STATE, HDD_ONLATCH, and HDD_OFFLATCH, and then drag these tags into the browser database.

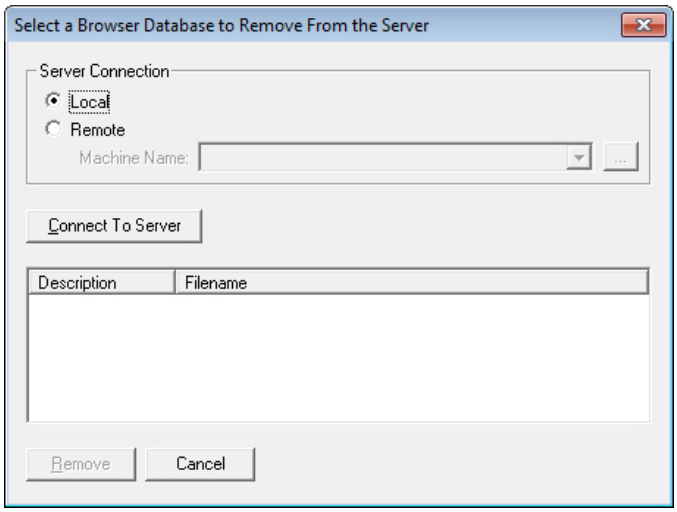
   Specify the module number (the module's position on the rack) and the point numbers as shown below.



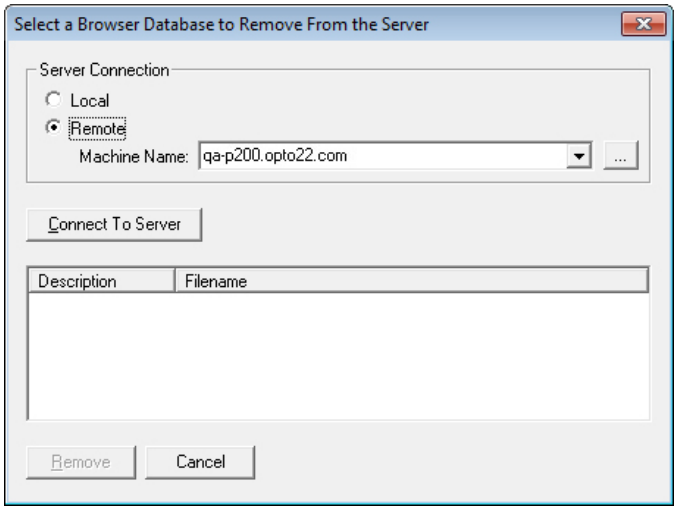
NOTE: When using high-density digital points, it is necessary to specify the module and points. You must provide a module number from 0–15, which corresponds to the module's position on the SNAP rack. For points, you have an option of indicating an individual point or a range or points.

REMOVING A BROWSER DATABASE FROM OPTOOPCSERVER

1. From the Opto Browser Configurator application window, choose Server > Remove Browser Database from Server.



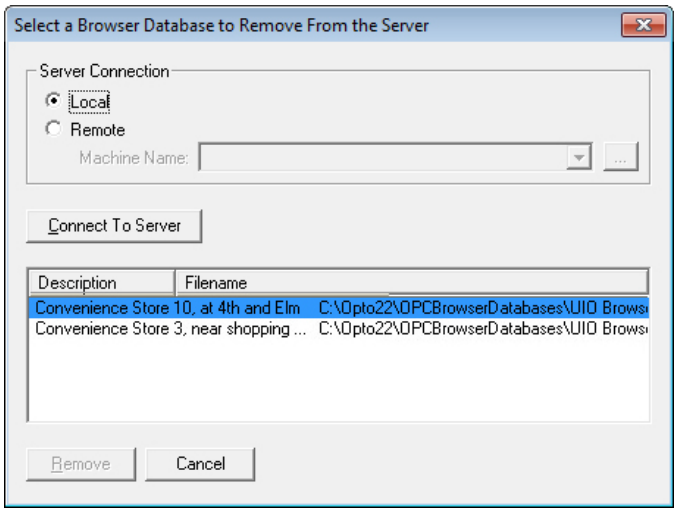
2. If OptoOPCServer resides on the same workstation as Opto Browser Configurator, choose Local. If OptoOPCServer resides on another computer on the network, choose Remote.



If you chose Remote, type the name of the computer running OptoOPCServer.

3. Click Connect to Server.

All browser databases loaded to OptoOPCServer are listed by their description.



4. Select the browser database you want to delete and click Remove.
5. Repeat Steps 1 through 4 to remove additional browser databases.

3: Configuring DCOM in Windows

Use this chapter to set up network security for OptoOPCServer using the Microsoft Windows Distributed Component Object Model (DCOM). When OptoOPCServer and OPC client applications reside on different computers, OptoOPCServer and OPC client applications use DCOM to communicate with each other. You need to have administrator privileges on both the OPC server and client computers.

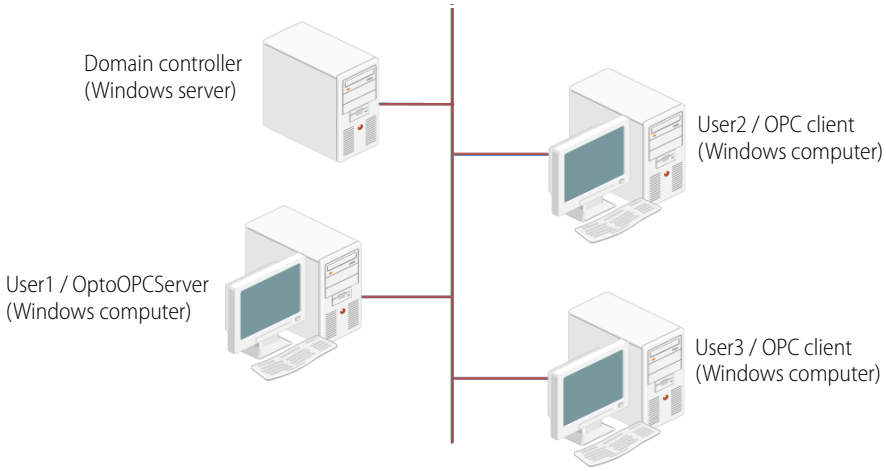
Is DCOM Required? DCOM is used only when OptoOPCServer and OPC client applications reside on different computers. *If your OptoOPCServer and OPC client applications reside on the same computer, DCOM is not used and you do not need the instructions in this chapter.*

NETWORK CONFIGURATION

Because the OptoOPCServer application and the OPC client are on different computers, you must configure DCOM in Windows on both the server and client computers. Windows configuration requirements depend on whether the computers are part of a domain network or a workgroup network.

Domain Network

In a *domain*, a domain controller manages the network, contains all user definitions, and provides network services. The user logs into a domain controller by providing a username and password. After logging in, the user has access to services provided by the network. OptoOPCServer resides on a user workstation and is one of these services.



Workgroup Network

A *workgroup* is a peer-to-peer network between computers with no domain controller. OptoOPCServer is usually on one computer and the OPC client applications on other computers.

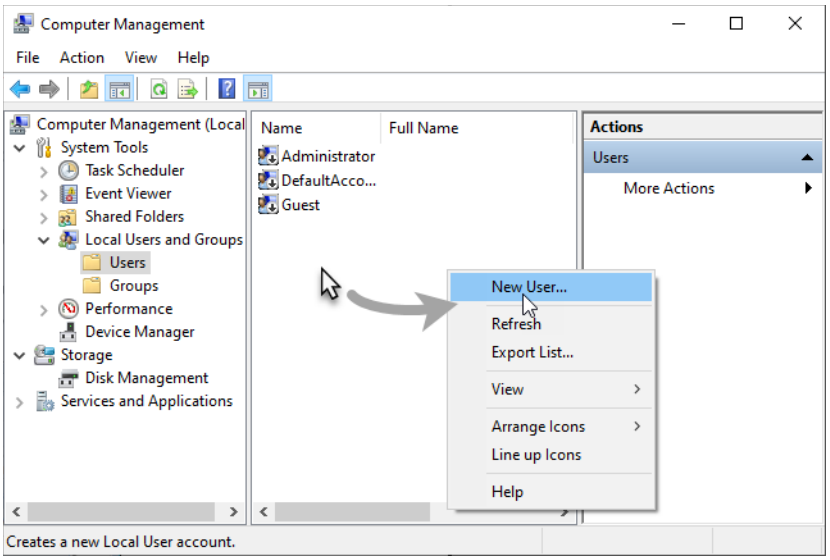
CONFIGURING WINDOWS ON THE OPTOOPCSERVER COMPUTER

Use these steps only if OptoOPCServer and the OPC client software are on separate computers.

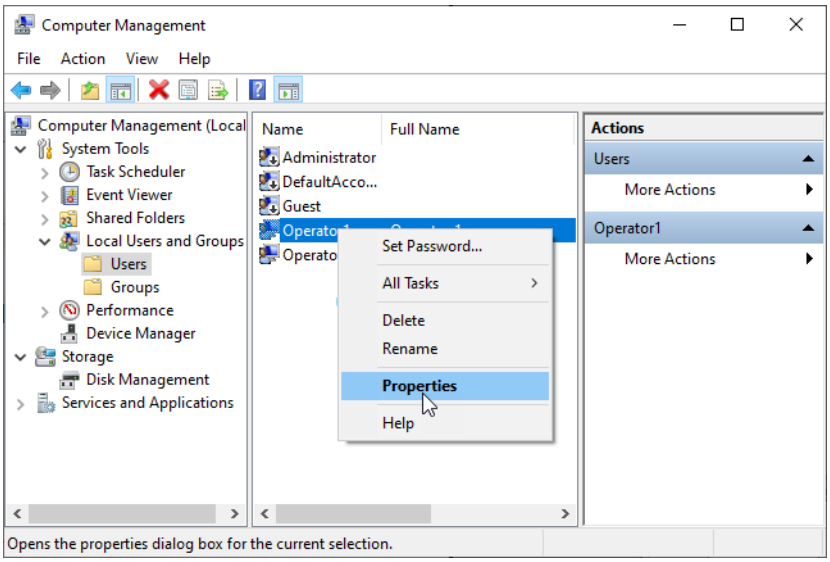
A. (Workgroup Only) Creating User Accounts on the OptoOPCServer Computer

If you have a workgroup, you need to create a user account on the OptoOPCServer computer for each client user account that will access the OptoOPCServer.

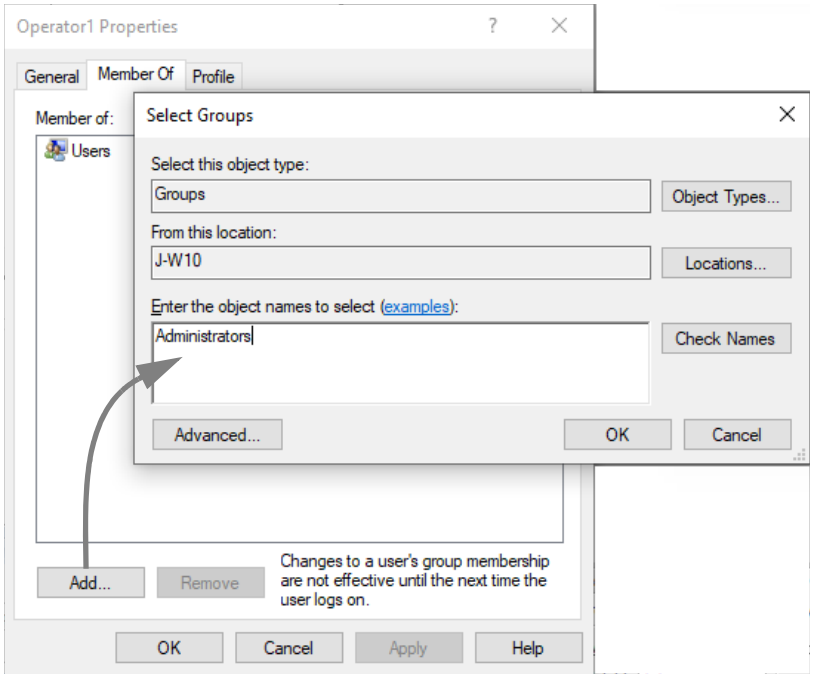
1. Right-click the Start menu and choose Computer Management.
   The Computer Management dialog box appears.
2. Expand Local Users and Groups, and then open the Users folder.
3. Place the cursor in the center panel's blank space, below the list of users; right-click and select New User.



4. In the New User dialog box, enter a name for the new user and complete other fields if you wish.
5. Type the password in the Password and Confirm Password fields, and then click Create.
6. Repeat steps 3-5 for each new user. When you have added all the new users, click Close.
7. In the list of users, right-click one of the new user names.



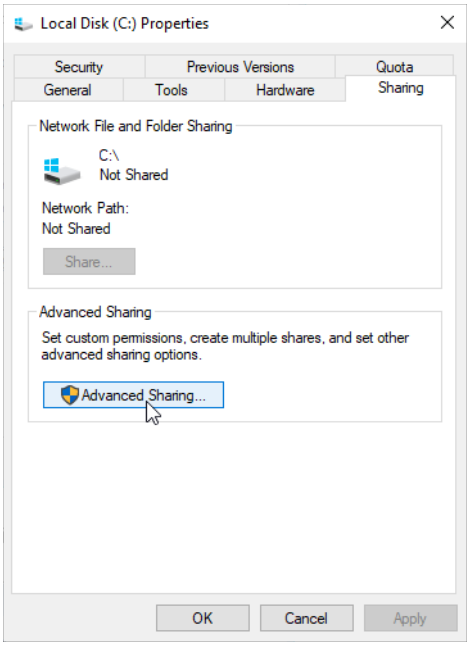
8. Choose Properties from the dropdown menu. In the Properties dialog box, click the Member Of tab, and then click the Add button.
9. In the Select Groups dialog box, type: `Administrators`



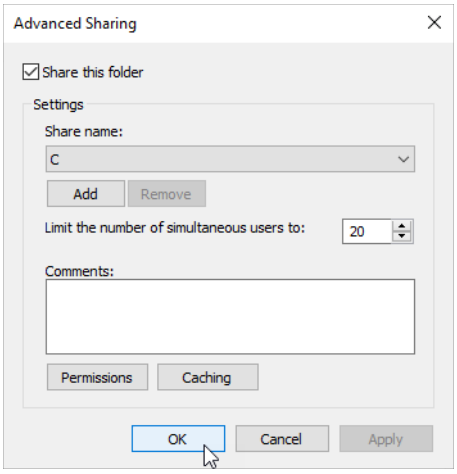
10. Click OK.
11. Repeat steps 7–10 on the OPC server workstation for each client user account that will access the OPC server.
12. Click OK to close the Properties dialog box.
13. Close the Computer Management window.

B. Enabling Network Sharing on the OPC Server Computer

1. Open Windows Explorer. Right-click the C: drive and select Give access to > Advanced Sharing.



2. In the Sharing tab, click the Advanced Sharing button.



3. Check Share this Folder, and then click OK.
4. Click Close.

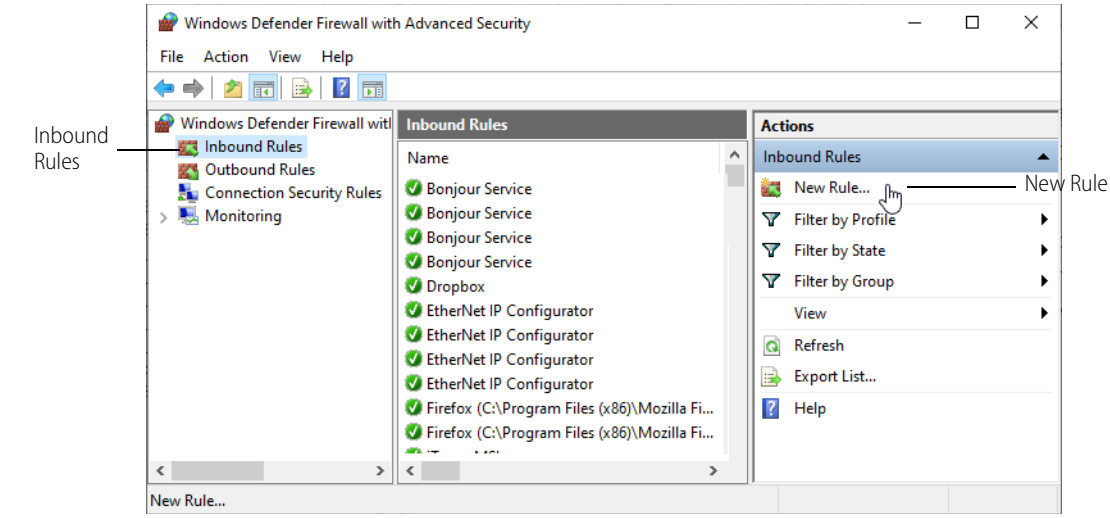
C. Configuring the Windows Firewall on the OptoOPCServer Computer

In this section you'll open port 135 and then configure the firewall to allow client programs to run.

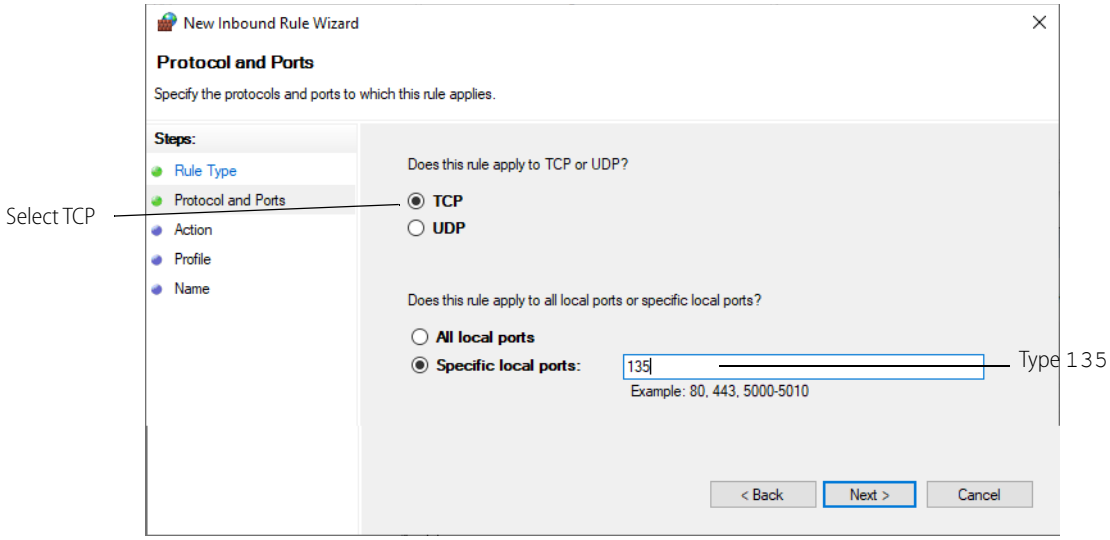
Opening Port 135 for Communications through the Windows Firewall

1. Click the Start button and start typing Control; click Control Panel.
2. Click Windows Defender Firewall.
3. In the left navigation, click Advanced settings. If you're prompted for an administrator password or confirmation, type the password or provide confirmation.

   The Windows Defender Firewall with Advanced Security dialog box opens.
4. In the left pane of the dialog box, click Inbound Rules.



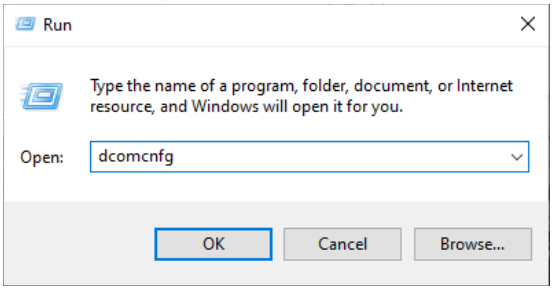
5. In the right pane, click New Rule to start the New Inbound Rule Wizard.
6. On the Rule Type screen, select Port, and then click Next.
7. On the Protocol and Ports screen, select TCP and enter the port number: `135`



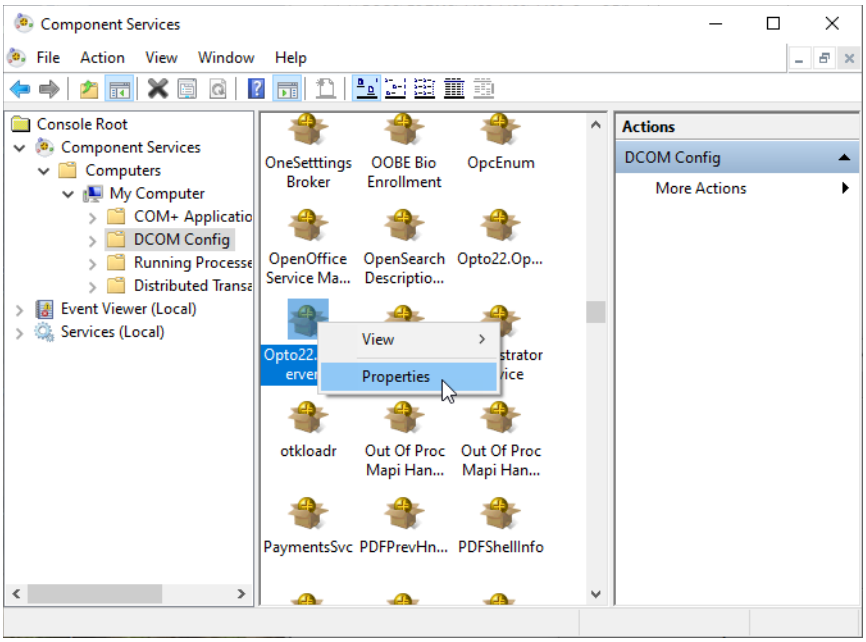
8. Cick Next.
9. On the Action screen, select Allow the Connection, and then click Next.
10. On the Profile screen, select Domain, Private, and Public, and then click Next.
11. On the Name screen, enter a name such as "Open Port 135," and then click Finish.
    In the Windows Defender Firewall dialog box, your new rule appears in the Action pane.
12. Close the Windows Defender Firewall dialog box.

D. Configuring DCOM Properties on the OptoOPCServer Computer

1. Select Start and type Run (or press the Windows Start key + R) to open the Run dialog box.
2. Type: `dcomcnfg`



3. Click OK.
4. In the Component Services window, expand the folders Component Services, Computers, and My Computer.
5. Select DCOM Config under My Computer.
   The applications with DCOM configurations appear in the middle pane.

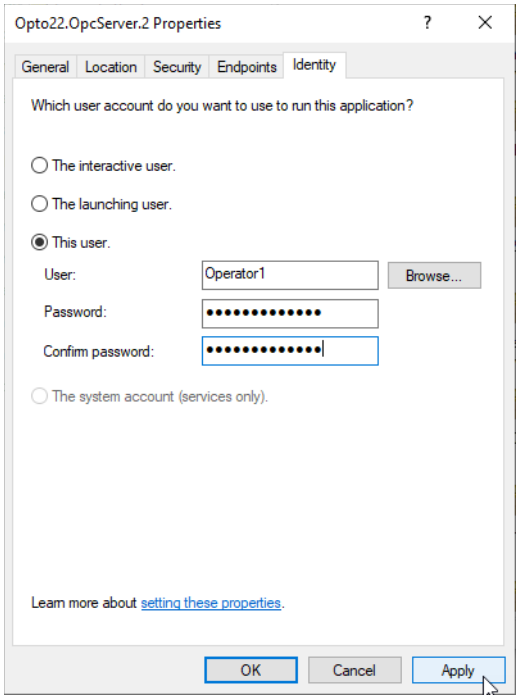


6. Scroll down to Opto22.OPCServer.2. Right-click Opto22.OPCServer.2 and choose Properties.
7. Click the Identity tab and select This User.
8. Enter the network account under which OptoOPCServer will run on this computer (must be a user account that already exists).

   NOTE: In a domain, the user must have a domain account with administrator privileges on the OptoOPCServer computer.

9. Type the password in the Password and Confirm Password fields.

   NOTE: If you change your domain password, you will need to change the password here as well.



10. Click Apply.

DCOM security settings for OptoOPCServer workstation

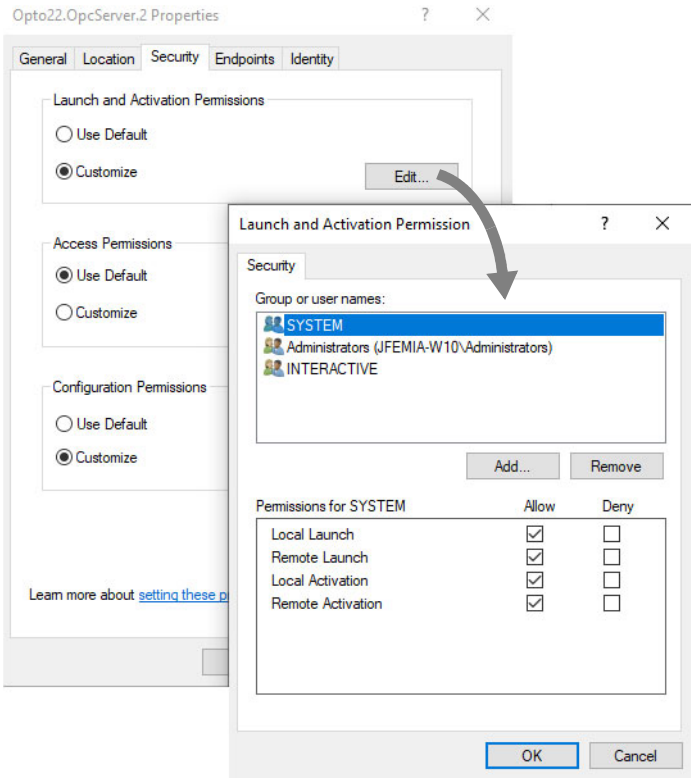
Use the following table as a checklist for the DCOM security settings for the OptoOPCServer workstation described in steps 2 through 6 below.

	Custom Opto22.OPCServer.2 Launch Permissions					Default Launch Permissions	
	Administrators	**INTERACTIVE**	**SYSTEM**	**Everyone**	**This User***	**Administrators**	**Everyone**
Local Launch	X	X	X	X	X	X	X
Remote Launch	X		X	X	X	X	X
Local Activation	X	X	X	X	X	X	X
Remote Activation	X		X	X	X	X	X

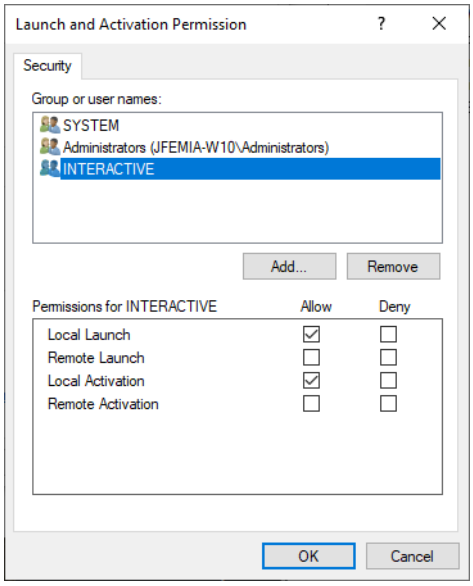
* *This User* is the network account under which OptoOPCServer will run, specified in step 8 on page 26.

1. Still in the Opto22.OpcServer.2 Properties dialog box, click the Security tab.

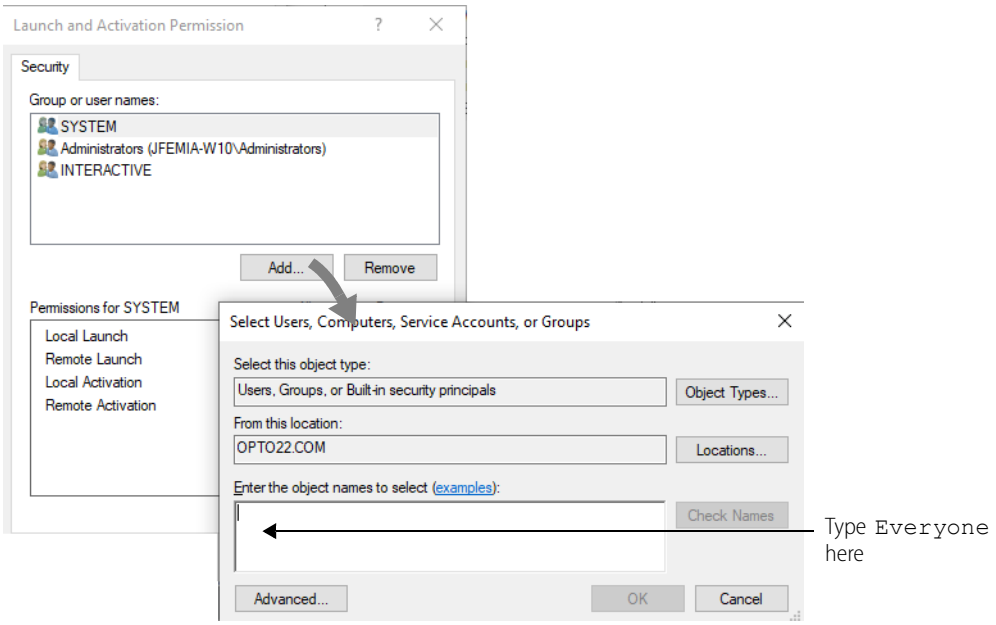
2. In the Security tab, select Customize under Launch and Activation Permissions, and then click Edit.



3. Select SYSTEM and check all of the Allow boxes.
4. Select Administrators and check all of the Allow boxes.
5. Select INTERACTIVE and check the Allow boxes for Local Launch and Local Activation.



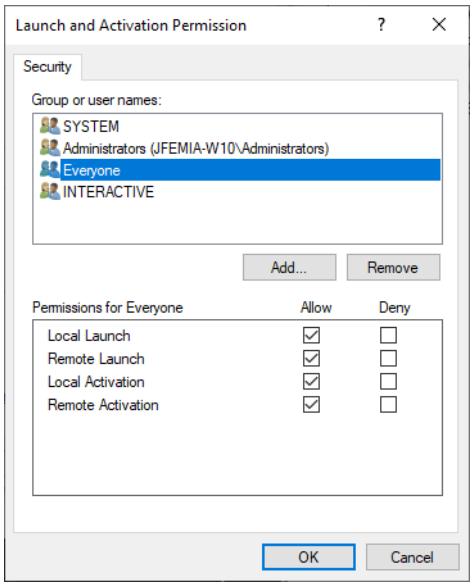
6. Click Add.



7. Type `Everyone`, and then click Check Names.

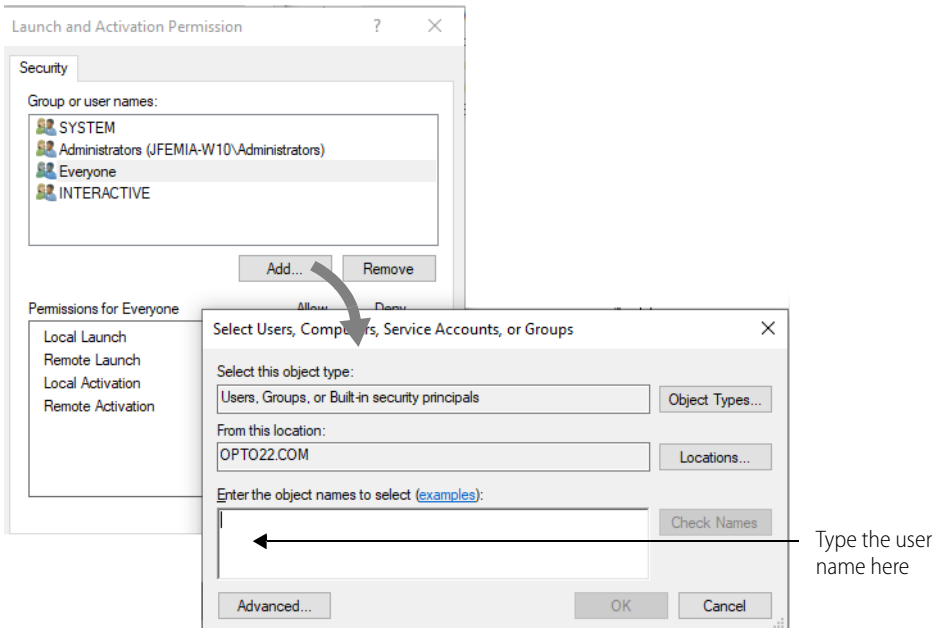
   `Everyone` is underlined.
8. Click OK to close the dialog box.

   `Everyone` is added to the list in the Security list.
9. Select Everyone and check all of the Allow boxes.



Now you'll add each user account that will be a client for OptoOPCServer. (For a workgroup, it's the users you created in the section on page 22). You are giving these accounts permission to launch or activate the OPC server if it is not already running.

10. Add each user account that will be a client for OPC Server.
    a. Click Add.



b. Type the name of the user.

c. Click Check Names.

The user name is underlined if correct.

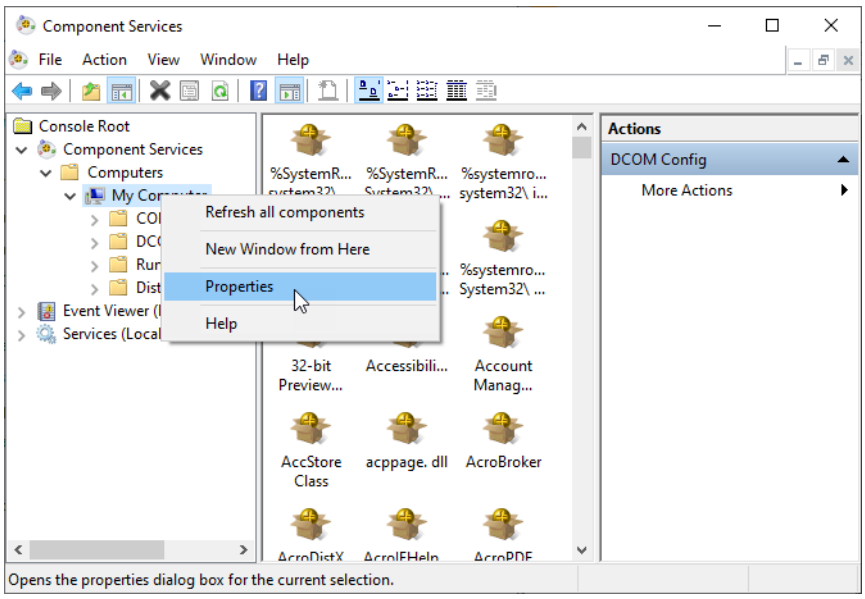
d. Click OK.

11. For each new user, select the new user, check all of the Allow boxes, and then click OK.

12. Click OK to close the Opto22.OPCServer.2 properties dialog box, but leave the Component Services window open.

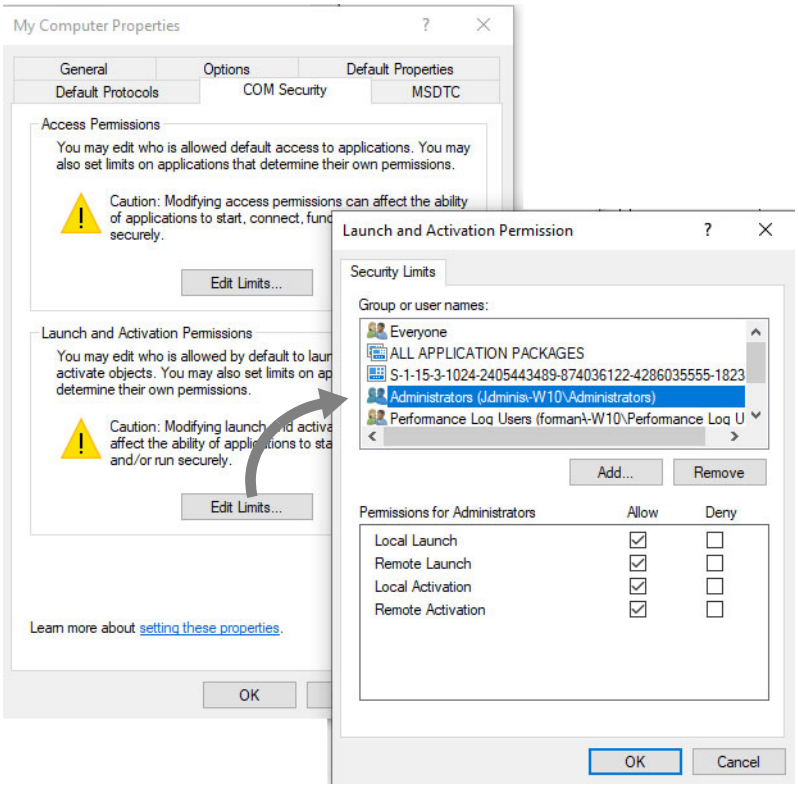
Default Launch Permissions for OptoOPCServer Workstation

1. In the Component Services left navigation pane, right-click My Computer and select Properties from the pop-up menu.



2. Click the COM Security tab.

3. In the Launch and Activation Permissions area, click Edit Limits.



4. Select Administrators, and check all of the Allow boxes.
5. Select Everyone, and check all of the Allow boxes.
6. Click OK to close the Launch Permission dialog box.

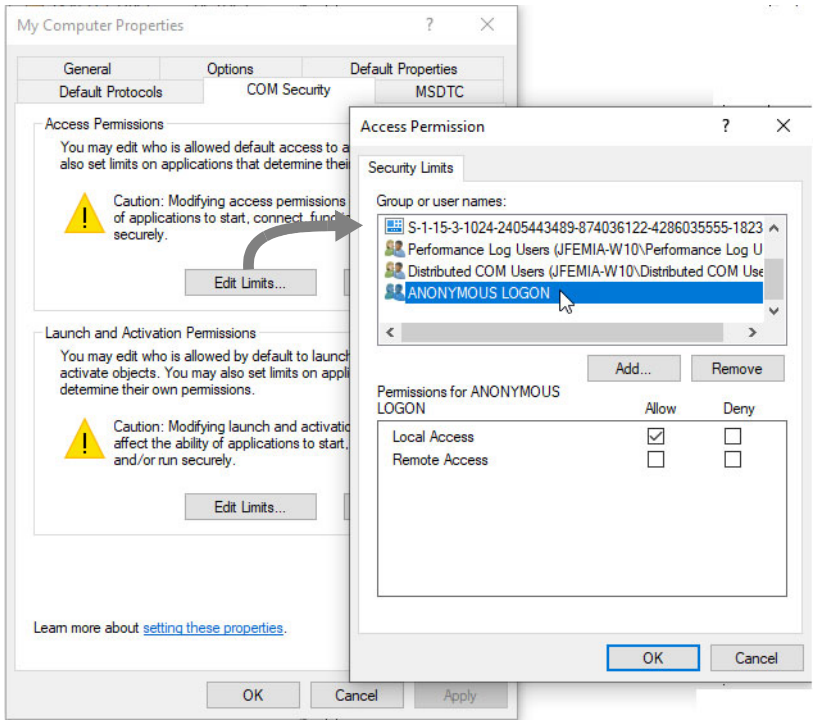
Default Access Permissions for OptoOPCServer Workstation

Use the following table as a checklist for the default COM security settings for the OptoOPCServer workstation described in steps 1 through b below.

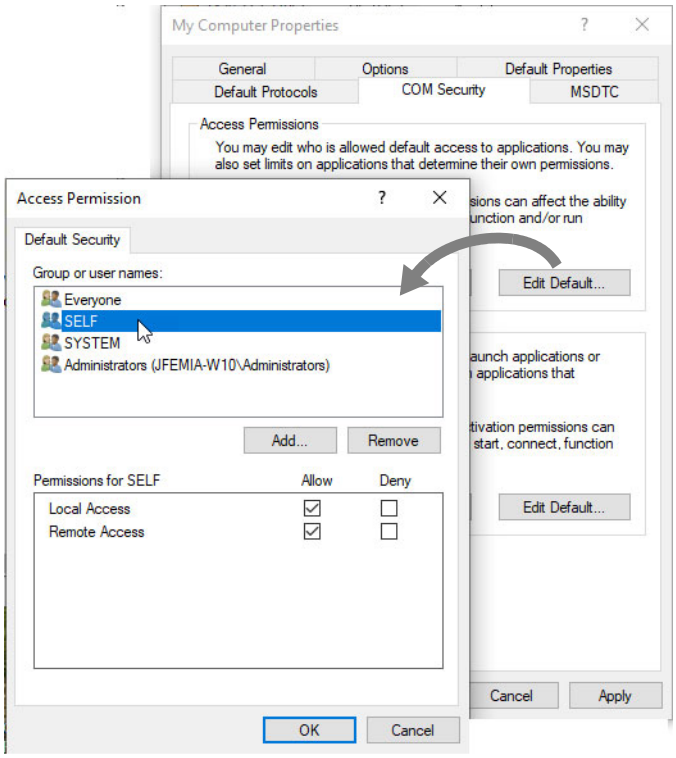
	Default COM Security Settings					
	Access Permissions: Limits		Access Permissions: Defaults			
	ANONYMOUS LOGON	Everyone	SELF	SYSTEM	This User*	Everyone
Local Access	X	X	X	X	X	X
Remote Access	X	X	X	X	X	X

* *This User* is the network account under which OptoOPCServer will run, specified in step 8 on page 26.

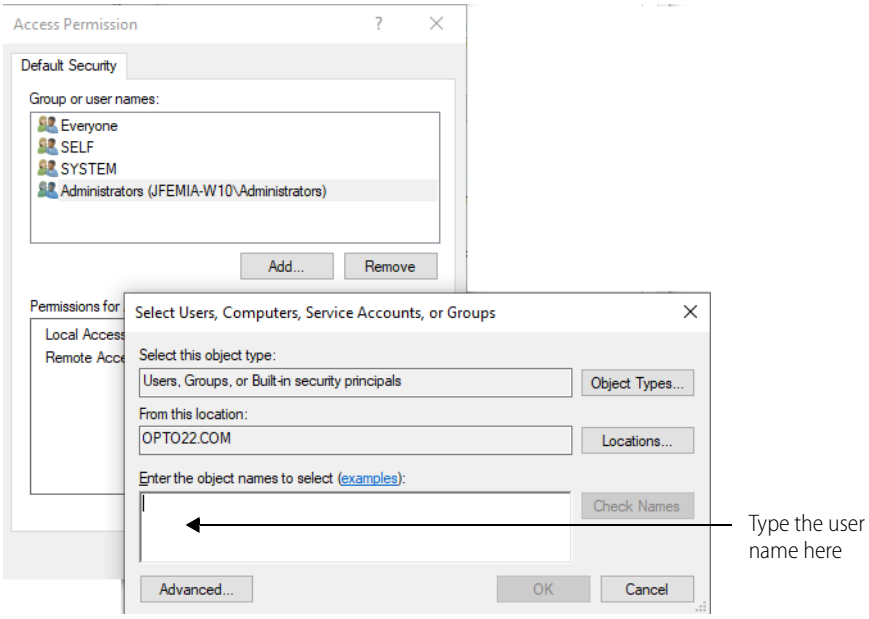
1. In the COM Security tab, under Access Permissions, click Edit Limits.



2. In the Access Permission dialog box, select ANONYMOUS LOGON and check both Allow boxes. Click OK.
3. Click Edit Limits again to open the Access Permission dialog box. Select Everyone, check both Allow boxes, and then click OK to close the Access Permission dialog box.
4. In the My Computer Properties window, under Access Permissions, click Edit Default.



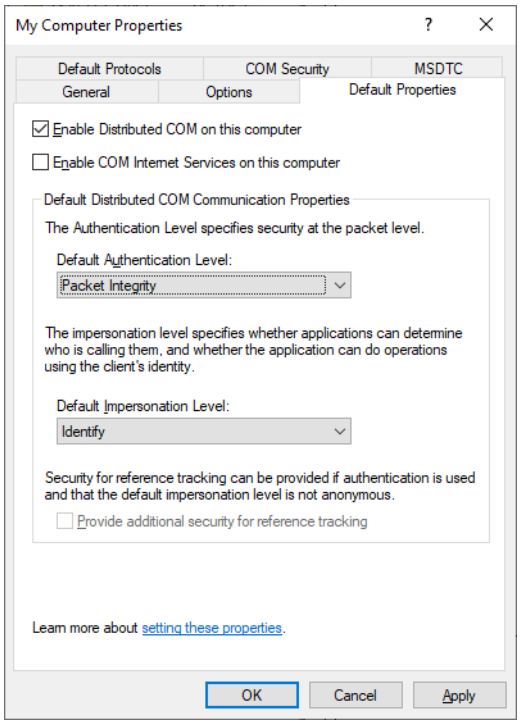
5. Select SELF and check both Allow boxes. (Don't click OK.)
6. Select SYSTEM and check both Allow boxes.
7. Select Everyone and check both Allow boxes.
8. Select the network account under which OptoOPCServer will run on this computer (specified in step 8 on page 26) and check both Allow boxes.
9. Add each user who will need to access OptoOPCServer:
   a. Click Add.



   b. Type the name of the user.
   c. Click Check Names. The workstation name and user name appear underlined.
   d. Click OK.
10. For each new user name added, check both Allow boxes.
11. Click OK to close the Access Permission dialog box.

Default Properties for the OptoOPCServer Workstation

1. In the My Computer Properties window, click the Default Properties tab and do the following:
   a. Select Enable Distributed COM on this computer.
   b. Do not select Enable COM Internet Services on this computer.
   c. Under Default Authentication Level, select Packet Integrity.
   d. Under Default Impersonation Level, select Identify.



2. Click Apply.
3. At the dialog box that opens saying your applications may not work when you do this, click Yes.
4. Click OK to close the dialog box, and then close the other open dialog boxes.

CONFIGURING WINDOWS ON THE OPC CLIENT COMPUTER

You can have one or more OPC client computers. For each client computer:

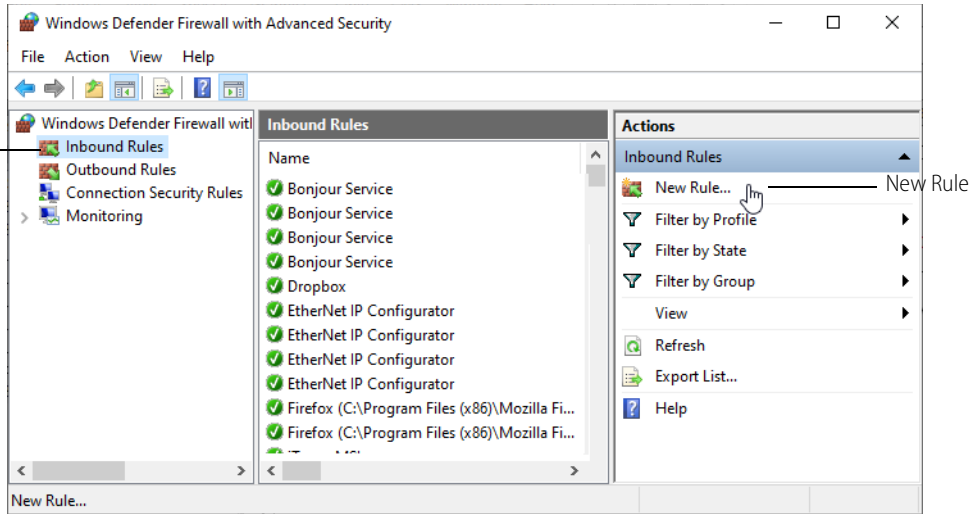
1. To provide the client-side components required by OptoOPCServer, install OptoOPCServer software on the client computer.
2. Follow steps in the rest of this section to configure the client computer.

A. Configuring the Windows Firewall on the OPC Client Computer

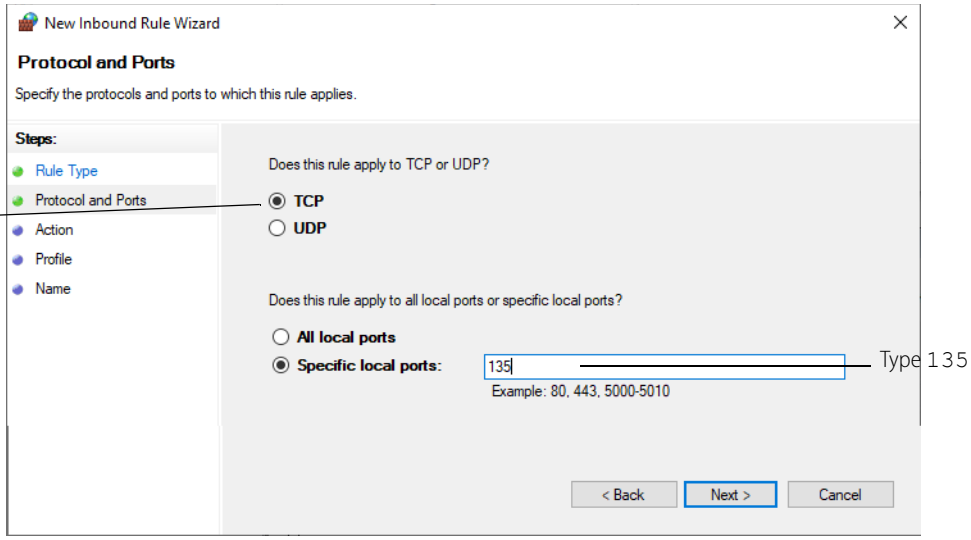
In this section you'll open port 135 and then configure the firewall to allow client programs to run.

1. Click the Start button and start typing Control; click Control Panel.
2. Click Windows Defender Firewall.
3. In the left navigation, click Advanced settings. If you're prompted for an administrator password or confirmation, type the password or provide confirmation.

   The Windows Defender Firewall with Advanced Security dialog box opens.
4. In the left pane of the dialog box, click Inbound Rules.



5. In the right pane, click New Rule to start the New Inbound Rule Wizard.
6. On the Rule Type screen, select Port, and then click Next.
7. On the Protocol and Ports screen, select TCP and enter the port number: `135`

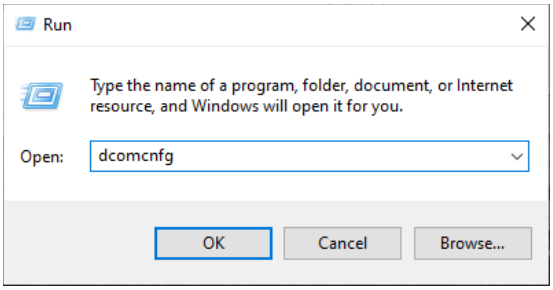


8. Cick Next.
9. On the Action screen, select Allow the Connection, and then click Next.
10. On the Profile screen, select Domain, Private, and Public, and then click Next.
11. On the Name screen, enter a name such as "Open Port 135," and then click Finish.
    Your new rule appears near the bottom of the Action pane.
12. Close the Advanced Security Window.

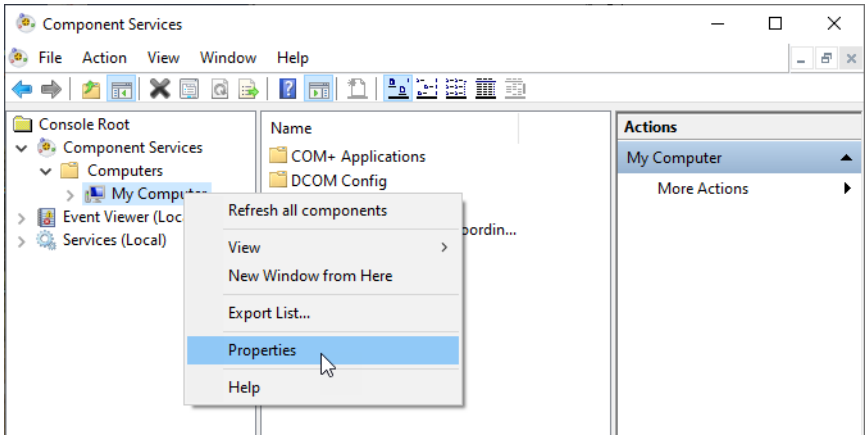
B. Configuring DCOM Properties on the OPC Client Computer

1. Select Start and type Run (or press the Windows Start key + R) to open the Run dialog box.

2. Type: `dcomcnfg`



3. Click OK.
4. In the Component Services window, expand the Component Services folder to see My Computer.
5. Right-click My Computer, and then select Properties from the pop-up menu.

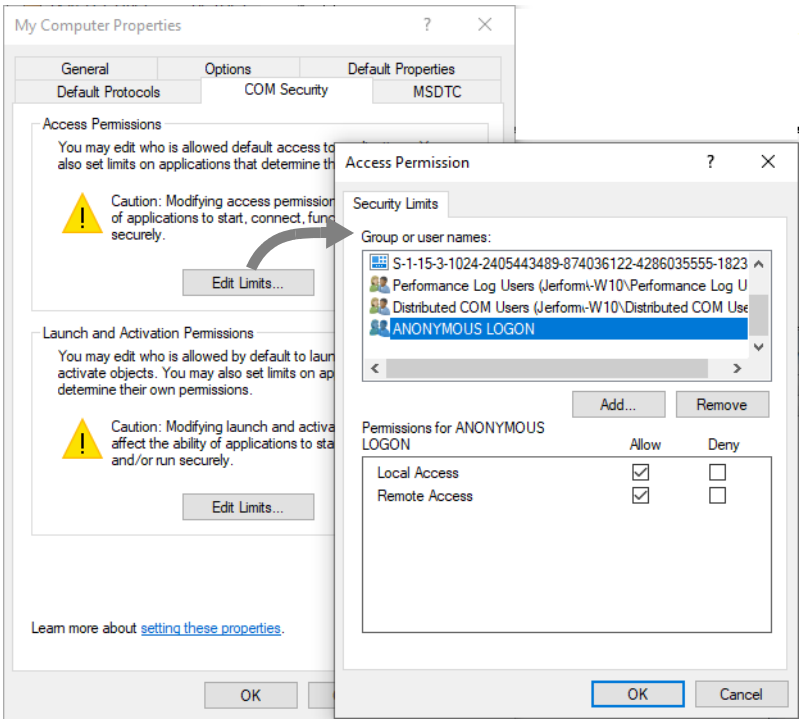


Use the following table as a checklist for the default COM security settings for the client workstation described in steps 6 through 13 below.

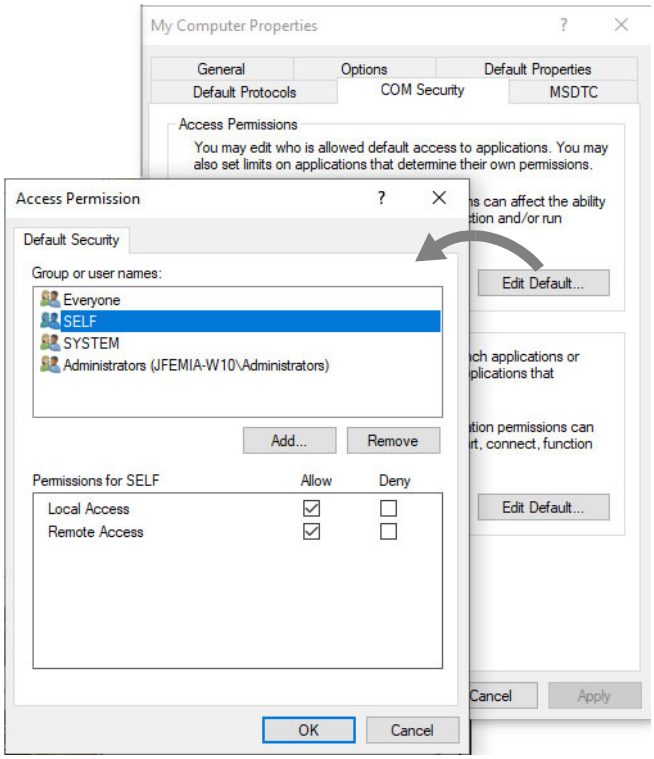
	COM Security Tab > Access Permissions				
	Edit Limits		Edit Defaults		
	ANONYMOUS LOGON	Everyone	SELF	SYSTEM	This User*
Local Access	X	X	X	X	X
Remote Access	X	X	X	X	X

* *This User* is the network account under which OptoOPCServer will run, specified in step 8 on page 26.

6. Click the COM Security tab. Under Access Permissions, click Edit Limits.



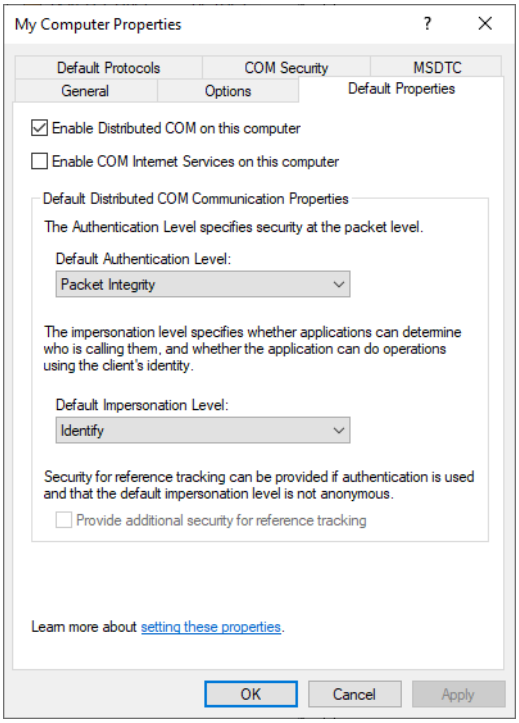
7. In the Access Permission dialog box, select ANONYMOUS LOGON and check both Allow boxes.
8. Select Everyone, check both Allow boxes, and then click OK to close the Access Permission dialog box.
9. In the My Computer Properties window, under Access Permissions, click Edit Default.



10. Select SELF and check both Allow boxes.
11. Select SYSTEM and check both Allow boxes.
12. Select the network account under which OptoOPCServer will run on its workstation (specified in step 8 on page 26) and check both Allow boxes.
13. Click OK to close the Access Permission dialog box.

Default Properties for the OptoOPCServer Client

1. In the My Computer Properties window, click the Default Properties tab and do the following:
   a. Select Enable Distributed COM on this computer.
   b. Make sure Enable COM Internet Services on this computer is *not* selected.
   c. In the Default Authentication Level drop-down list, choose Packet Integrity.
   d. Under Default Impersonation Level, choose Identify.

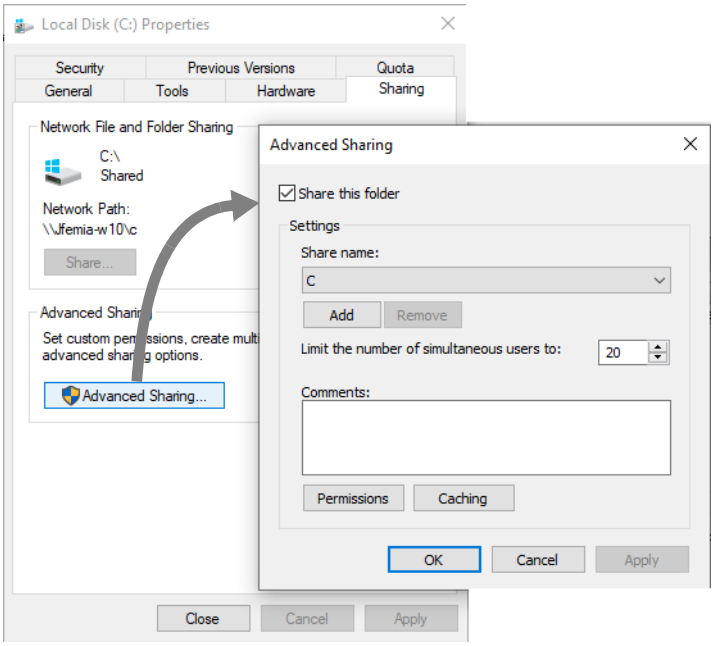


2. Click Apply.
3. At the dialog box that opens saying your applications may not work when you do this, click Yes.
4. Click OK to close the My Computer Properties dialog box, and then close the other open dialog boxes.

C. Enabling Network Sharing on the OPC Client Computer

1. Open Windows Explorer. Right-click the C: drive and choose Give access to > Advanced Sharing.

2. In the Properties window, make sure you're in the Sharing tab, and then click the Advanced Sharing button.



3. Select Share this Folder, and then click OK.
4. Close the Local Disk Properties dialog box.

4: Opto 22 OPC Item Definitions

An OPC item definition consists of two parts: the *access path* and the *item ID*. OptoOPCServer does not use the access path and therefore ignores it, but it employs two item ID syntaxes: one to access memory map data from an Opto 22 brain or controller, and another to access control strategy data from an Opto 22 controller.

To construct an item ID to access **memory map data**, see the next section "Item ID Syntax for Accessing Memory Map Data."

To construct an item ID to access **control strategy data**, see "Item IDs for Accessing Control Strategy Data."

ITEM ID SYNTAX FOR ACCESSING MEMORY MAP DATA

Item IDs for accessing memory map data are constructed using the general form:

```
[Device|Protocol|Location]ItemName[PointNumber]
```

Use the tables and examples starting on page 44 to determine how to construct specific item IDs.

Here's what the parameters mean:

Parameter	Value	Remarks
Device:	mmio	Opto 22 memory-mapped device, including *groov* EPIC processors, SNAP PAC R-series controllers, EB-series brains, SNAP Ethernet brains, and E1 and E2 brain boards
Protocol:	ip	Network protocol
Location:	`tcp:` followed by the IP address in dot notation, and port number of the *groov* EPIC processor, SNAP PAC or SNAP Ultimate brain.	Example: `tcp:10.192.55.81:2001` The default port number is 2001 for all memory map data and is separated from the IP address by a colon. Instead of the IP address, you can also use the device's hostname (DNS).
ItemName:	See the tables starting on the next page for the ItemName to use for what you want to do.	The type of data to read or write *NOTE: Reading an invalid float value (for example, reading engineering units on a digital point) returns a value of -99999.*
PointNumber:	The IO unit point number (0–63 or 0–4096) or the Scratch Pad element to read or write to	For high-density digital points, you provide a Module Number and a Point Number.

Table Column Headings

Here are descriptions of some of the table column headings in the tables that start on page 44:

- **Data Type**: The data types Bool, Long, Float, and Str have enumerated VARTYPE values of VT_BOOL, VT_I4, VT_R4, and VT_BSTR, respectively.
- **Action**: The Action column shows whether the item name can be read, written to, or both.
- **Scanned**: Item names marked with a "No" in the Scanned column are not continually scanned. Consequently, an OPC client will not receive subscription data for these ItemNames. To read data for these ItemNames, an OPC client must perform an explicit read using the Async Read or Sync Read from Device commands.
- **Elements**: For I/O point ItemNames, the Elements column shows the I/O unit point numbers available, either 0–63 for the old memory map areas or 0–4095 for the expanded areas. Scratch Pad ItemNames show the number of elements that can be read from and written to a Scratch Pad memory area.
- **Array Allowed?**: ItemNames marked with a "Yes" in the Array Allowed? column can be accessed as multiple elements in an array.

  Example: COUNTERDATA[60-63] returns counter data for I/O points 60 through 63.

ITEM NAME COMPATIBILITY

Opto 22's OptoOPCServer provides a substantial amount of functionality for an OPC client, some of which is not available to typical third-party applications incorporating an OPC client. Most third-party applications are either based on SCADA (supervisory control and data acquisition) or are HMIs (human machine interfaces) that do not allow the end-user to use all of OptoOPCServer's available functionality.

Most SCADAs and HMI environments rely only on scanned (or polled) tags. In the Item Name tables that start on page 44, a *Yes* in the Scanned column indicates that a tag is supported by applications encapsulating an OPC client. For a tag marked *No* (for not scanned), in order for it to work your software package must have the ability to support a *synchronous OPC read. This is because* Opto 22's embedded intelligent I/O features (*such as* latches, events, counters, and pulse measurement) require a clearing or enable/disable of the feature to work correctly; and the clearing or reset requires the access to a *synchronous OPC read* Item Name. It is likely that your third-party software application does not have a synchronous OPC read capability. However, to find out for sure, contact the software manufacturer.

Here is an example of this limitation: One of the intelligent I/O features in Opto 22's architecture is an *On-Latch*. When a digital input transitions from an off-state to an on-state, the On-Latch register is turned on. The register will remain on until either a clear or a read-and-clear command is issued. The clearing or the read-and-clear are only via the synchronous OPC read (non-scanned) tags. For applications that do not contain the synchronous OPC read functionality, the application will never be able to clear the latch.

The limitations of the third-party software may be overcome with some redesign of the system. The changes are dependent upon the overall automation system architecture.

For systems with an Opto 22 controller such as a *groov* EPIC or SNAP PAC controller, Opto 22 recommends adding strategy logic to use table elements or variables (which are scanned items) to trigger the synchronous read operations within the PAC Control strategy. PAC Control's command set contains equivalent clear and read-and-clear functionality.

If your system has only I/O units with Ethernet brains (such the SNAP PAC EB-series) and no Opto 22 controllers, we recommended changing the EB-series brains to equivalent SNAP PAC R-series controllers. Create a minimal PAC Control strategy to provide scannable interfaces through table elements or variables to trigger the synchronous OPC read functions.

USING MMIO AND CONT SCANNER TAGS TO GET I/O DATA

In order to meet the needs of different types of Ethernet network configurations, OptoOPCServer offers *direct* and *indirect* methods of getting I/O data.

- **Direct (MMIO)** This is the preferred method to use when the controller and the I/O unit are both on the same Ethernet network as the OptoOPCServer computer. This method provides the best performance because the MMIO scanner scans the I/O units directly rather than through the control engine. The CONT scanner can also be used, but it is slower because it scans I/O tags indirectly through the controller. For the OPC Item ID, use "MMIO" and TCP port 2001, which is the default port for Ethernet brains and the I/O side of R1 and R2 controllers.
- **Indirect (CONT)** Use this method when the OptoOPCServer computer has access only to the controller's host port and does not have access to the I/O unit. The OptoOPCServer CONT (controller) scanner gets the data indirectly through the controller to the I/O unit. For the OPC Item ID, use "CONT" and TCP port 22001, which is the controller's host port.

USING HOSTNAMES IN OPC ITEM IDS

The OptoOPCServer supports the use of *hostnames* as a substitute for the IP address in the Item ID. For both the I/O (MMIO) and PAC Control (CONT) tag scanners, you may use a DNS resolved name or a static host name in ETC/HOSTS. The HOSTS file is typically located at C:\WINDOWS\SYSTEM32\DRIVERS\ETC\

Here are examples of how the hostname would appear in the Item Names. Also see the next section, "Using MMIO and CONT Scanner Tags to Get I/O Data."

Example 1: Using the Device's IP Address

```
[mmio|ip|tcp:10.192.55.10:2001]EU[12-36]
[mmio|ip|tcp:10.192.55.10:2001]SECS_SINCE_POWERUP
[cont|ip|tcp:10.192.55.10:22001]I32;VALUE;i32MyIntegerVariable
[cont|ip|tcp:10.192.55.10:22001]I32T;VALUE[0-249];i32tblIntegerTable
```

Example 2: Using the Device's DNS or Hostname

```
[mmio|ip|tcp:EB2:2001]EU[12-36]
[mmio|ip|tcp:EB2:2001]SECS_SINCE_POWERUP
[cont|ip|tcp:EPICPR1:22001]I32;VALUE;i32MyIntegerVariable
[cont|ip|tcp:SNAPPACS2:22001]I32T;VALUE[0-249];i32tblIntegerTable
```

For OEMs that have a common OPC Client configuration but have to locally change the IP Address, using a hostname allows the OPC Client configuration to remain constant and a single location where the IP address may be remapped. In the above example, hostname EB2's IP address may be changed without changing the OPC Client configuration.

For system integrators that wish to easily configure an office test configuration before deploying the actual system, the modification of the hostname lookup address is very simple. If IP addresses were to be used, the entire Item Name configuration would have to be changed (a very tedious and lengthy process).

Opto 22 recommends using a *static* hostname as this is both the simplest method and the least error prone. The HOSTS file is typically located at C:\WINDOWS\SYSTEM32\DRIVERS\ETC\

ANALOG POINT

Syntax

```
[Device|Protocol|Location]ItemName[PointNumber]
```

ItemName

Use an ItemName from the following table:

	To do this	Use this ItemName	Data Type	Action	Scanned	Elements	Array Allowed?
Expanded* Memory Map Area	Read or write current engineering units value	EU_4096	Float	R/W	Yes	0–4095	No
	Read or write current counts value	COUNTS_4096	Float	R/W	Yes	0–4095	No
	Read current minimum value	MIN_4096	Float	R	Yes	0–4095	No
	Read current maximum value	MAX_4096	Float	R	Yes	0–4095	No
	Read then clear minimum value	MIN_READCLEAR_4096	Float	R	No	0–4095	No
	Read then clear maximum value	MAX_READCLEAR_4096	Float	R	No	0–4095	No
Old* Memory Map Area	Read or write current engineering units value	EU	Float	R/W	Yes	0–63	Yes
	Read or write current counts value	COUNTS	Float	R/W	Yes	0–63	Yes
	Read current minimum value	MIN	Float	R	Yes	0–63	Yes
	Read current maximum value	MAX	Float	R	Yes	0–63	Yes
	Read then clear minimum value	MIN_READCLEAR	Float	R	No	0–63	Yes
	Read then clear maximum value	MAX_READCLEAR	Float	R	No	0–63	Yes

* The old memory map area supports modules with up to four points per module. The expanded memory map area supports high-density modules as well as modules with four points per module.

PointNumber

To calculate the PointNumber, use the following formula:

```
PointNumber=(nModule * nPointsPerModule) + nPoint
```

where:

nModule=module number: 0-63, corresponding to the module on the rack. *groov* EPIC chassis and SNAP PAC racks have up to 16 module positions (0-15).

nPointsPerModule=points per module: 4 (for old memory map area) or 64 (for expanded memory map tags such as eu_4096)

nPoint=module point number: 0-63 (for expanded map area) or 0-3 (for old memory map area)

Examples

To do this	Use this
Read or change the engineering units of analog module 2, point 1 using the expanded memory map	[mmio|ip|tcp:10.192.55.1:2001]eu_4096[129]
Read or change the engineering units of analog module 2, point 1 using the old memory map	[mmio|ip|tcp:10.192.55.1:2001]eu[9]
Read or write counts of analog points at module 3, points 0 and 1 using the old memory map	[mmio|ip|tcp:10.192.55.1:2001]counts[12-13]

DIGITAL POINT FOR 4-CHANNEL MODULES

Syntax

`[Device|Protocol|Location]ItemName[PointNumber]`

ItemName

Use an ItemName from the following table:

To do this	Use this ItemName	Data Type	Action	Scanned	Elements	Array Allowed?
Read or write current on/off state	STATE	Bool	R/W	Yes	0–63	No
Read on-latch state	ONLATCH	Bool	R	Yes	0–63	No
Read off-latch state	OFFLATCH	Bool	R	Yes	0–63	No
Read or write state of active counter flag*	ACTIVECOUNTER	Bool	R/W	Yes	0–63	No
Read current counter value*	COUNTERDATA	Long	R	Yes	0–63	Yes
Read then clear on-latch state	ONLATCH_READCLEAR	Bool	R	No	0–63	Yes
Read then clear off-latch state	OFFLATCH_READCLEAR	Bool	R	No	0–63	Yes
Read then clear current counter value	COUNTERDATA_READCLEAR	Long	R	No	0–63	Yes
Read the value of an on-pulse measurement	ONPULSE	Int32	R	Yes	0-63	Yes
Inspect the status of an on-pulse measurement	ONPULSE_STATUS	Int32	R	Yes	0-63	No
Read the value of an off-pulse measurement	OFFPULSE	Int32	R	Yes	0-63	Yes
Inspect the status of an Off-Pulse measurement	OFFPULSE_STATUS	Int32	R	Yes	0-63	No
Read the pulse frequency	FREQUENCY	Int32	R	Yes	0-63	Yes
Read the pulse period	PERIOD	Int32	R	Yes	0-63	Yes
Read the period status	PERIOD_STATUS	Int32	R	Yes	0-63	No
Read the on-time totalizer	ONTIME_TOTALIZER	Int32	R	Yes	0-63	Yes
Read the off-time totalizer	OFFTIME_TOTALIZER	Int32	R	Yes	0-63	Yes

* To use a digital input point as a counter, you must configure it as a counter, and the counter must be active (that is, the active counter flag is set to true). For more information on counters, see the user's guide for the device.

PointNumber

To calculate the PointNumber, use the following formula:

```
PointNumber=(nModule * nPointsPerModule) + nPoint
```

where:

nModule=module number: 0-63, corresponding to the module on the rack. *groov* EPIC chassis and SNAP PAC racks have up to 16 module positions (0-15).

nPointsPerModule=points per module: 4 (for old memory map area) or 64 (for expanded memory map tags such as eu_4096)

nPoint=module point number: 0-63

Examples

To do this	Use this
Read or change the state of digital point at module 1, point 2 (on a 4-point module)	[MMIO|ip|tcp:10.192.55.1:2001]state[6]
Read and clear the counter of digital input point at module 2, point 3 (on a 4-point module)	[MMIO|ip|tcp:10.192.55.1:2001]counterdata_readclear[11]

DIGITAL POINT FOR TPO AND PULSING

Syntax `[Device|Protocol|Location]ItemName[PointNumber]`

ItemName Use an ItemName from the following table:

To do this	Use this ItemName	Data Type	Action	Scanned	Elements	Array Allowed?
Read or write the TPO percentage	TPO_PERCENT	Float	R/W	YES	0-511	No
Read or write the TPO period	TPO_PERIOD	Float	R/W	YES	0-511	No

PointNumber To calculate the PointNumber, use the following formula:

```
PointNumber=(nModule * 32) + nPoint
```

where:

nModule=module number: 0-15, corresponding to the module on the rack.

nPoint=module point number: 0-31

Example

To do this	You would use this item ID
Read or write the TPO percentage for point 9 on module 3 (using the Expanded Memory Map Area)	[mmio|ip|tcp:10.192.55.1:2001]TPO_PERCENT[105]

HIGH-DENSITY DIGITAL (HDD) POINT

Syntax `[Device|Protocol|Location]ItemName(Module)[Point]`

ItemName Use an ItemName from the following table:

To do this	Use this ItemName	Data Type	Action	Scanned	Module	Point	Array Allowed?
Read or write current on/off state of point 'p' of HDD module 'm'	HDD_STATE(m)[p]	Bool	R/W	Yes	0–15	0–31	No
Read on-latch state of point 'p' of HDD module 'm'	HDD_ONLATCH(m)[p]	Bool	R	Yes	0–15	0–31	No
Read off-latch state of point 'p' of HDD module 'm'	HDD_OFFLATCH(m)[p]	Bool	R	Yes	0–15	0–31	No
Clear on-latch state of point 'p' of HDD module 'm' [1]	HDD_ONLATCH_CLEAR(m)[p]	Bool	W	No	0–15	0–31	No
Clear off-latch state of point 'p' of HDD module 'm' [1]	HDD_OFFLATCH_CLEAR(m)[p]	Bool	W	No	0–15	0–31	No
Read current counter value of point 'p' of HDD module 'm'	HDD_COUNTER(m)[p]	Long	R	Yes	0–15	0–31	No
Read then clear current counter value of point 'p' of HDD module 'm' [2]	HDD_COUNTER_READCLEAR(m)[p]	Long	R	Yes	0–15	0–31	No
Read or write current on/off state for all 32 points of HDD module 'm' [3]	HDD_BANK_STATE(m)	Long	R/W	Yes	0–15	n/a	No
Read current on-latch state for all 32 points of HDD module 'm' [3]	HDD_BANK_ONLATCH(m)	Long	R	Yes	0–15	n/a	No
Read current off-latch state for all 32 points of HDD module 'm' [3]	HDD_BANK_OFFLATCH(m)	Long	R	Yes	0–15	n/a	No

To do this	Use this ItemName	Data Type	Action	Scanned	Module	Point	Array Allowed?
Clear current on-latch state for all 32 points of HDD module 'm' [1, 4]	HDD_BANK_ONLATCH_CLEAR(m)	Long	W	No	0–15	n/a	No
Clear current off-latch state for all 32 points of HDD module 'm' [1, 4]	HDD_BANK_OFFLATCH_CLEAR(m)	Long	W	No	0–15	n/a	No

[1] Notice this action is write-only. Write a value of “true” to clear the respective latch. This is not a "Read and Clear"; it is more like a "Write to Clear."
[2] This action is a "Read and Clear" (unlike the latches).
[3] The value sent is a 32-bit mask where “1” bits equal On and “0” bits equal Off.
[4] The value sent is a 32-bit mask where “1” bits equal "Clear respective latch" and “0” bits equal "Do nothing."

Examples

To do this	You would use this item ID
Read the current module state as a Boolean of module 3, point 7.	[mmio|ip|tcp:10.192.55.1:2001]HDD_STATE(3)[7]
Read the bank state of module 15 as a bitmask.	[mmio|ip|tcp:10.192.55.1:2001]HDD_BANK_STATE(15)
Read the counter on point 9 of HDD module 6	[mmio|ip|tcp:10.192.55.1:2001]HDD_COUNTER(6)[9]
Read or write all up to 32 states on an HDD module installed in position 15	[mmio|ip|tcp:10.192.55.1:2001]HDD_BANK_STATE(15)

POINT CONFIGURATION

Syntax

`[Device|Protocol|Location]ItemName[PointNumber]`

ItemName

Use an ItemName from the following table:

	To do this	Use this ItemName	Data Type	Action	Scanned	Elements	Array Allowed?
Expanded* Memory Map Area	Read module configuration type	MODULETYPE_4096	Int32	R	Yes	0–4095	No
	Read or write point configuration type	POINTTYPE_4096	Int32	R/W	Yes	0–4095	No
	Read or write point feature	FEATURE_4096	Int32	R/W	Yes	0–4095	No
	Read or write offset value (analog calibration)	OFFSET_4096	Float	R/W	Yes	0–4095	No
	Read or write gain value (analog calibration)	GAIN_4096	Float	R/W	Yes	0–4095	No
	Read or write analog high scaling factor	HISCALE_4096	Float	R/W	Yes	0–4095	No
	Read or write analog low scaling factor	LOSCALE_4096	Float	R/W	Yes	0–4095	No
	Read or write an expanded area analog input hi-clamp value	HICLAMP_4096	Float	R/W	Yes	0-4095	Yes
	Read or write an expanded area analog input lo-clamp value	LOCLAMP_4096	Float	R/W	Yes	0-4095	Yes
Old" Memory Map Area	Read module configuration type	MODULETYPE	Int32	R	Yes	0–63	No
	Read or write point configuration type	POINTTYPE	Int32	R/W	Yes	0–63	No
	Read or write point feature	FEATURE	Int32	R/W	Yes	0–63	No
	Read or write offset value (analog calibration)	OFFSET	Float	R/W	Yes	0–63	No
	Read or write gain value (analog calibration)	GAIN	Float	R/W	Yes	0–63	No
	Read or write analog high scaling factor	HISCALE	Float	R/W	Yes	0–63	No
	Read or write analog low scaling factor	LOSCALE	Float	R/W	Yes	0–63	No

* The old memory map area supports modules with up to four points per module. The expanded memory map area supports high-density modules as well as modules with four points per module.

PointNumber

To calculate the PointNumber, use the following formula:

```
PointNumber=(nModule * nPointsPerModule) + nPoint
```

where:

nModule=module number: 0-63, corresponding to the module on the rack. *groov* EPIC chassis and SNAP PAC racks have up to 16 module positions (0-15).

nPointsPerModule=points per module: 4 (for old memory map area) or 64 (for expanded memory map tags such as eu_4096)

nPoint=module point number: 0-63

Examples

To do this	You would use this item ID
Read the module type installed in position 7	[mmio|ip|tcp:10.192.55.1:2001]MODULETYPE[7]
Read or write the high scaling parameter on point 7 of high-density analog module 11	[mmio|ip|tcp:10.192.55.1:2001]HISCALE_4096[455]

ALARM

Syntax

```
[Device|Protocol|Location]ItemName[PointNumber]
```

ItemName

Use an ItemName from the following table:

To do this	Use this ItemName	Data Type	Action	Scanned	Elements	Array Allowed?
Set high alarm state	ALARM_HI_STATE	Bool	R	Yes	0–63	No
Enable high alarm	ALARM_HI_ENABLE	Bool	R/W	Yes	0–63	No
Set high alarm setpoint	ALARM_HI_SETPOINT	Float	R/W	Yes	0–63	No
Set high alarm deadband	ALARM_HI_DEADBAND	Float	R/W	Yes	0–63	No
Set low alarm state	ALARM_LO_STATE	Bool	R	Yes	0–63	No
Enable low alarm	ALARM_LO_ENABLE	Bool	R/W	Yes	0–63	No
Set low alarm setpoint	ALARM_LO_SETPOINT	Float	R/W	Yes	0–63	No
Set low alarm deadband	ALARM_LO_DEADBAND	Float	R/W	Yes	0–63	No

PointNumber

To calculate the PointNumber, use the following formula:

```
PointNumber=(nModule * nPointsPerModule) + nPoint
```

where:

nModule=module number: 0-63, corresponding to the module on the rack. *groov* EPIC chassis and SNAP PAC racks have up to 16 module positions (0-15).

nPointsPerModule=points per module: 4 (for old memory map area) or 64 (for expanded memory map tags such as eu_4096)

nPoint=module point number: 0-63

Examples

To do this	You would use this item ID
Enable or disable the high alarm on point 4	[mmio|ip|tcp:10.192.55.1:2001]ALARM_HI_ENABLE[4]

SCRATCH PAD

NOTE: groov EPIC processors, SNAP PAC controllers, SNAP PAC Ethernet brains, and Ultimate brains only.

Syntax `[Device|Protocol|Location]ItemName[Index]`

ItemName Use an ItemName from the following table:

To do this	Use this ItemName	Data Type	Action	Scanned	Index	Array Allowed?
Read or write Scratch Pad bit	SP_BIT	Bool	R/W	Yes	0–63	Yes
Read or write Scratch Pad 32-bit integer	SP_INTEGER	Long	R/W	Yes	0–1023	Yes
Read or write Scratch Pad 32-bit integer	SP_INTEGER_EXT	Long	R/W	Yes	1024–10239	Yes
Read or write Scratch Pad float	SP_FLOAT	Float	R/W	Yes	0–1023	Yes
Read or write Scratch Pad float	SP_FLOAT_EXT	Float	R/W	Yes	1024–10239	Yes
Read or write Scratch Pad string	SP_STRING	Str	R/W	Yes	0–63	Yes

NOTE: Scratch pad 64-bit integer is not supported via OPC.

Examples

To do this	You would use this item ID
Read or write 32-bit integer scratch pads, indexes 103 through 335	[mmio|ip|tcp:10.192.55.1:2001]SP_INTEGER[103-335]
Read or write float scratch pad register 11	[mmio|ip|tcp:10.192.55.1:2001]SP_FLOAT[11]
Read or write extended float scratch pad registers 9,000 through 10,239	[mmio|ip|tcp:10.192.55.1:2001]SP_FLOAT_EXT[9000-10239]

PID

Syntax `[Device|Protocol|Location]ItemName[PointNumber]`

ItemName Use an ItemName from the following table:

To do this	Use this ItemName	Data Type	Action	Scanned	PID Index	Array Allowed?
Current value: Input	PID_CV_IN	F	R	Y	0–127	N
Current value: Setpoint	PID_CV_SP	F	R	Y	0–127	N
Current value: Output	PID_CV_OUT	F	R/W	Y	0–127	N
Current value: Feed Forward	PID_CV_FF	F	R/W	Y	0–127	N
Current value: Error	PID_CV_ERROR	F	R	Y	0–127	N
Current value: Gain contribution	PID_CV_P	F	R	Y	0–127	N
Current value: Integral contribution	PID_CV_I	F	R	Y	0–127	N
Current value: Derivative contribution	PID_CV_D	F	R	Y	0–127	N
Current value: Integral	PID_CV_INTEGRAL	F	R	Y	0–127	N
Last scanned value: Input	PID_LSV_IN	F	R/W	Y	0–127	N
Last scanned value: Setpoint	PID_LSV_SP	F	R/W	Y	0–127	N
Status flags	PID_STATUS	I	R/W	Y	0–127	N
Status flags on mask	PID_STATUS_ON	I	W	Y	0–127	N
Status flags off mask	PID_STATUS_OFF	I	W	Y	0–127	N
Tuning: Proportional value	PID_TUNE_P	F	R/W	Y	0–127	N

To do this	Use this ItemName	Data Type	Action	Scanned	PID Index	Array Allowed?
Tuning: Integral value	PID_TUNE_I	F	R/W	Y	0–127	N
Tuning: Derivative value	PID_TUNE_D	F	R/W	Y	0–127	N
Tuning: Feed Forward gain	PID_TUNE_FF	F	R/W	Y	0–127	N
Configuration: Max output change allowed]	PID_CFG_MAX_OUT	F	R/W	Y	0–127	N
Configuration: Min output change allowed	PID_CFG_MIN_OUT	F	R/W	Y	0–127	N
Configuration: Scan time in seconds	PID_CFG_SCAN_TIME	F	R/W	Y	0–127	N
Configuration: Output when input is low	PID_CFG_LOW_RANGE	F	R/W	Y	0–127	N
Configuration: Output when input is high	PID_CFG_HI_RANGE	F	R/W	Y	0–127	N
Configuration: Algorithm choice	PID_CFG_ALG	I	R/W	Y	0–127	N
Configuration: Manual mode - 1=Yes 0=No	PID_CFG_MAN_MODE	I	R/W	Y	0–127	N
Configuration: Flags	PID_CFG_FLAGS	I	R/W	Y	0–127	N
Configuration: Flags on mask	PID_CFG_FLAGS_ON	I	W	Y	0–127	N
Configuration: Flags off mask	PID_CFG_FLAGS_OFF	I	W	Y	0–127	N
Configuration: Input mem map address	PID_CFG_MM_IN	I	R/W	Y	0–127	N
Configuration: Setpoint mem map address]	PID_CFG_MM_SP	I	R/W	Y	0–127	N
Configuration: Output mem map address	PID_CFG_MM_OUT	I	R/W	Y	0–127	N
Scaling: Input low range	PID_SCALE_IN_LOW	F	R/W	Y	0–127	N
Scaling: Input high range	PID_SCALE_IN_HI	F	R/W	Y	0–127	N
Scaling: Output lower clamp	PID_SCALE_OUT_LOW	F	R/W	Y	0–127	N
Scaling: Output upper clamp	PID_SCALE_OUT_HI	F	R/W	Y	0–127	N
Scan Counter	PID_SCAN_COUNTER	I	R/W	Y	0–127	N

Examples:

To do this	Use this
Read the current input on PID index 38	[mmio|ip|tcp:10.192.55.1:2001]PID_CV_IN[38]
Read or write the proportional value on PID index 11	[mmio|ip|tcp:10.192.55.1:2001]PID_TUNE_P[11]

DIAGNOSTICS

Syntax `[Device|Protocol|Location]ItemName`

ItemName Use an ItemName from the following table:

To do this	Use this ItemName	Data Type	Action	Scanned	PID Index	Array Allowed?
Read the firmware version installed on the I/O unit. A hexadecimal number.	FW_VERSION	Int32	R	Yes	N/A	No
Read the IO unit type. A hexadecimal number.	UNIT_TYPE	Int32	R	Yes	N/A	No
Read the IO unit description.	UNIT_DESC	String	R	Yes	N/A	No
Read or write the reported temperatures (zero for Celsius, 1 for Fahrenheit).	DEGREES_F_OR_C	Int32	R/W	Yes	N/A	No
Read the scanner flags. A hexadecimal number.	SCANNER_FLAGS	Int32	R	Yes	N/A	No

To do this	Use this ItemName	Data Type	Action	Scanned	PID Index	Array Allowed?
Read the ARCNET scan counter.	ARC_SCAN_COUNTER	Int32	R	Yes	N/A	No
Read the Ethernet late collisions counter.	ENET_LATE_COLL	Int32	R	Yes	N/A	No
Read the Ethernet excessive collisions counter.	ENET_EXCESS_COLL	Int32	R	Yes	N/A	No
Read the Ethernet other errors counter.	ENET_ERROR_OTHER	Int32	R	Yes	N/A	No
Read the Smart Modules bit mask.	SMART_MODULES	Int32	R	Yes	N/A	No
Read the ARCNET reconfiguration counter.	ARCNET_RECON	Int32	R	Yes	N/A	No
Read the ARCNET reconfiguration initiated by modules counter.	ARCNET_RECON_IO	Int32	R	Yes	N/A	No
Read the number of seconds the IO unit has been up.	SECS_SINCE_POWERUP	Int32	R	Yes	N/A	No
Read the milliseconds the IO unit has been up.	MS_SINCE_POWERUP	Int32	R	Yes	N/A	No
Read the milliseconds the IO unit has been powered up as a 64-bit value.	MS_SINCE_POWERUP64	Int32	R	Yes	N/A	No
Read the Ethernet MAC reset counter.	ENET_MAC_RESET	Int32	R	Yes	N/A	No
Read the digital output reset counter.	DIG_OUT_RESET	Int32	R	Yes	N/A	No
Read the digital interrupt failure counter.	DIG_INT_FAILS	Int32	R	Yes	N/A	No
Read the ARCNET transmission counter.	ARCNET_TX	Int32	R	Yes	N/A	No
Read the ARCNET acknowledge counter.	ARCNET_ACKS	Int32	R	Yes	N/A	No
Read the ARCNET timeout counter.	ARCNET_TIMEOUTS	Int32	R	Yes	N/A	No
Read the ARCNET other error counter.	ARCNET_OTHER	Int32	R	Yes	N/A	No
Read the ARCNET timeout value.	ARCNET_TIMEOUT_VAL	Int32	R	Yes	N/A	No
Read the ARCNET receive counter.	ARCNET_RX_INTS	Int32	R	Yes	N/A	No
Read the ARCNET scanner time.	ARC_SCAN_TIME	Int32	R	Yes	N/A	No
Read the standard digital scanner time.	STDDIG_SCAN_TIME	Int32	R	Yes	N/A	No
Read the module discovery counter.	ARCNET_MOD_DISC(m)	Int32	R	Yes	N/A	Yes
Write a command to the IO unit.	MMP_WRITE_CMD*	Int32	R/W	Yes	N/A	No
Read the IO unit date-time as a string.	RTC_DATETIME_STR	String	R/W	Yes	N/A	No
Read and write the maximum ARCNET scan time.	MAX_ARC_SCAN_TIME	Float	R/W	Yes	N/A	No
Rand and write the maximum ARCNET scan time.	DIG_FEAT_SCAN_TIME	Float	R/W	Yes	N/A	No
Read the IO unit primary (ENET1) MAC address.	PRIMARY_MAC_ADDR	Int32	R	Yes	0-1	Yes
Read and write the communication watchdog timeout.	COMM_WATCHDOG_TIME	Int32	R/W	Yes	N/A	No

* MMP_WRITE_CMD writes a value to the Mem Map address F038 0000. For more information on the values you can send, see “Status Area Write—Read/Write” in Appendix A of form 1465, *OptoMMP Protocol Guide*. For example, to store all configuration data to flash, you would use a value of 3.

Examples:

To do this	Use this
Read how long in seconds, the IO Unit or Controller has been powered up	[mmio|ip|tcp:10.192.55.1:2001]SECS_SINCE_POWERUP

To do this	Use this
Report the number of Ethernet late collisions detected on the IO unit	[mmio|ip|tcp:10.192.55.1:2001]ENET_LATE_COLL
Report the number of times all of the IO modules have reset on the rack	[mmio|ip|tcp:10.192.55.1:2001]ARCNET_MOD_DISC(0-15)

SNMP SYSTEM STRINGS

Syntax

`[Device|Protocol|Location]ItemName`

ItemName

Use an ItemName from the following table:

To do this	Use this ItemName	Data Type	Action	Scanned	Array Allowed?
Read and write the SNMP System Name	SNMPSYSNAME	String	R/W	Yes	No
Read and write the SNMP System Location	SNMPSYSLOCATION	String	R/W	Yes	No
Read and write the SNMP System Contact	SNMPSYSCONTACT	String	R/W	Yes	No

Examples

To do this	You would use this item ID
Report the SNMP System Name	[mmio|ip|tcp:10.192.55.1:2001]SNMPSYSNAME

ITEM IDS FOR ACCESSING CONTROL STRATEGY DATA

Use the following format to construct item IDs for accessing variables in strategies running on *groov* EPIC processors, SNAP PAC controllers, and SNAP Ethernet-based controllers (for example, the SNAP-UP1-ADS):

`[Device|Protocol|Location]DataType;Property;VariableName`

Use the tables and examples starting on page 54 to determine how to construct specific item IDs.

NOTE: It's faster for OptoOPCServer to access I/O point data directly from the I/O unit using the mmio scanner. However, you must go through the controller (by using the cont scanner) when the I/O point data is on:

- *Serial I/O units connected to a controller*
- *Ethernet I/O units connected to a controller's secondary Ethernet port*
  ***EXCEPTION**: When the OptoOPCServer has a network connection to the secondary Ethernet network that the I/O units are on, you can access the points directly using the mmio scanner.*

Here's what these parameters mean:

Parameter	Value	Description
Device	cont	Opto 22 SNAP PAC or SNAP Ultimate I/O control engine or
	oc	OptoControl controllers with Ethernet
Protocol	ip	Network protocol
Location	`tcp:` followed by the IP address in dot notation, and port number of the *groov* EPIC processor, SNAP PAC or SNAP Ultimate brain.	Example: `tcp:10.192.55.81:22001` The default port number is 22001 for all memory map data and is separated from the IP address by a colon. Instead of the IP address, you can also use the device's hostname (DNS). Use port 2001 for an OptoControl controller.

Parameter	Value	Description
DataType	See the tables that follow for the DataType to use for what you want to do.	The type of strategy data to read from or write to the control engine.
Property		The value, state, or other property of the DataType.
VariableName		The name of the strategy variable, chart, or I/O unit.

***CAUTION**: Accessing I/O Item Names of remote serial I/O units can be very slow and is likely to cause performance issues. Similarly, systems with high volumes of Ethernet-based I/O tags accessed through a processor or controller may also have performance issues. To avoid such issues, consider using "MMIO" tags to directly access the I/O data.*

For more information on how to achieve efficient I/O data throughput, see the Optimizing PAC Project System Performance Technical Note (form 1776).

Table Column Headings

Note the following things in the tables starting on page 54:

- The Action column shows whether the item name can be read, written to, or both.
- Items marked with a "No" in the Scanned column are not continually scanned. Consequently, an OPC client will not receive subscription data for these items. To read data for these items, an OPC client must perform an explicit read using the Async Read or Sync Read from Device commands.

Using Pointers

You can get values for PAC Control strategy pointer variables by pre-pending PTR_ to the datatype being pointed to (for example, PTR_I32). You can use pointers for all data types except CTRLR.

INTEGERS, FLOATS, STRINGS, AND TIMERS

Syntax

```
[Device|Protocol|Location]DataType;Property;VariableName
```

DataType and Property

Use a DataType and Property from the following table:

To do this	Use this DataType	Available Properties	Action	Scanned
Integers				
Read or write 32-bit integer variable	I32[1]	VALUE	R/W	Yes
Read or write 64-bit integer variable	I64[2]	VALUE	R/W	Yes
Floats				
Read or write floating-point variable	F	VALUE	R/W	Yes
Strings				
Read or write string variable	S[3]	VALUE	R/W	Yes
Timers				
Read or write timer variable	T	VALUE	R/W	Yes

[1] In a 32-bit integer, add a period and the bit number after Value For example, "I32;Value.2;nPointsMask" reads bit 2 in the integer variable nPointsMask.
[2] If you want a 64-bit integer variable you must write your own custom client.
[3] A string variable cannot have embedded nulls.

Examples

To do this	Use this
Read or write the integer variable nStartFlag	[CONT|ip|tcp:10.192.55.1:22001]I32;Value;nStartFlag
Read or write bit 2 in the integer variable nPointsMask	[CONT|ip|tcp:10.192.55.1:22001]I32;Value.2;nPointsMask
Read or write the floating-point variable nPointsMask	[CONT|ip|tcp:10.192.55.1:22001]F;Value;fTankTemperature
Read or write the string variable sPartDescription	[CONT|ip|tcp:10.192.55.1:22001]S;Value;sPartDescription
Read or write the timer variable nOvenTime	[CONT|ip|tcp:10.192.55.1:22001]T;Value;nOvenTime

TABLES

Syntax

`[Device|Protocol|Location]DataType;Property;VariableName`

DataType and Property

Use a DataType and Property from the following table:

To do this	Use this DataType	Available Properties	Action	Scanned
Read or write one or more elements in a 32-bit integer table variable	I32T	VALUE	R/W	Yes
Read or write one or more elements in a 64-bit integer table variable	I64T[2]	VALUE	R/W	Yes
Read or write one or more elements in a floating-point table variable	FT	VALUE	R/W	Yes
Read or write one or more elements in a string table variable[1]	ST	VALUE	R/W	Yes

[1] A string table element cannot have embedded nulls.
[2] If you want a 64-bit integer table variable you must write your own custom client.

Notes:

- If you want to read a single bit from a 32-bit integer, add a period and the bit number after the table element number. For example, “I32T;Value[3].2;ntTotalWidgets” reads bit 2 of element 3 in the integer table variable ntTotalWidgets.
- To read or write multiple elements, add the range of elements after “Value.” For example, “I32T;Value[1-3];ntTotalWidgets” reads elements 1 through 3 in the integer table ntTotalWidgets. This is more efficient that using separate Item IDs for each table element.

Examples

To do this	Use this
Read or change the 32-bit integers in elements 6 through 11 of the integer table variable ntRetryLimits.	[CONT|ip|tcp:10.192.55.1:22001]I32T;Value[6-11];ntRetryLimits

STRATEGY CHARTS, I/O UNITS, AND CONTROLLERS

Syntax

```
[Device|Protocol|Location]DataType;Property;VariableName
```

DataType and Property

Use a DataType and Property from the following table:

To do this	Use this DataType	Available Properties	Action	Scanned
Start or stop a chart in a control strategy	CHART	START STOP	W	No
Suspend a running chart in a control strategy, or re-start a suspended chart	CHART	SUSPEND CONTINUE	W	No
Change the state of a chart in a control strategy	CHART	STATE*	R/W	Yes
Enable or disable from the controller to an I/O unit defined in a control strategy	IOUNIT	ENABLE	R/W	Yes
Read the following control engine information:				
• Available memory		MEMORY		
• Current time of the controller's clock		TIME		
• Current date of the controller's clock		DATE		
• Name of control strategy in the controller	CTRLR	STRATEGYNAME	R	No
• Total number of messages in the controller's message queue		ERRORCOUNT		
• Description of the latest message in the controller's message queue		LASTERROR		
• Time stamp of the strategy file on the controller		STRATEGYTIME		
• Date stamp of the strategy file on the controller		STRATEGYDATE		
Synchronize the controller time to the time on the computer that's running OptoOPCServer.	CTRLR	SYNCDATETIME	W	No

* Reading chart states will return the following values: 1 = stopped; 2 = suspended; 3 = running.
To change chart states, use the following parameters: 1 = stop; 2 = suspend; 3 = start; 4 = continue.

Examples

To do this	Use this
Suspend the running chart Process_Monitor	[CONT|ip|tcp:10.192.55.1:22001]CHART;Suspend;Process_Monitor

ANALOG POINTS

***CAUTION**: Be careful when using I/O Item Names through the PAC Control engine. Accessing I/O Item Names of remote serial I/O units is very slow and will likely cause severe performance issues. Also, systems with high volumes of Ethernet based I/O tags accessed through the controller may also have performance issues. In these cases, consider using "MMIO" tags to directly access the I/O data and circumvent accessing these values through the PAC Controller.*

For more information on how to achieve efficient I/O data throughput, see the Optimizing PAC Project System Performance Technical Note (form 1776).

Syntax

```
[Device|Protocol|Location]DataType;Property;VariableName
```

DataType and Property

Use APOINT as the DataType, and use a Property from the following table:

To do this	Use this DataType	Available Properties	Action	Scanned
Read or write analog values in engineering units to or from an I/O unit	APOINT	EU	R/W	Yes
Retrieve the lowest value of a specified analog input since its last reading	APOINT	MIN	R	Yes
Retrieve the lowest value of a specified analog input since its last reset, then reset it to the current value	APOINT	MIN_GET_CLEAR	R/W	No
Retrieve the peak value of a specified analog input since its last reading	APOINT	MAX	R	Yes
Retrieve the peak value of a specified analog input since its last reset, then reset it to the current value	APOINT	MAX_GET_CLEAR	R	No

Examples

To do this	Use this
Read analog point engineering units of point named fTankLevel	[CONT|ip|tcp:10.192.55.1:22001]Apoint;EU;fTankLevel

DIGITAL POINTS (ALL I/O UNITS)

***CAUTION**: Be careful when using I/O Item Names through the PAC Control engine. Accessing I/O Item Names of remote serial I/O units is very slow and will likely cause severe performance issues. Also, systems with high volumes of Ethernet based I/O tags accessed through the controller may also have performance issues. In these cases, consider using "MMIO" tags to directly access the I/O data and circumvent accessing these values through the PAC Controller.*

For more information on how to achieve efficient I/O data throughput, see the Optimizing PAC Project System Performance Technical Note (form 1776).

Syntax

```
[Device|Protocol|Location]DataType;Property;VariableName
```

DataType and Property

Use DPOINT as the DataType, and use a Property from the following table:

To do this	Use this DataType	Available Properties	Action	Scanned
Read a standard digital input counter value	DPOINT	COUNTER	R	Yes
Reactivate a standard digital input counter or quadrature counter	DPOINT	COUNTER_ENABLE	W	No
Read and clear a standard digital input counter or quadrature counter value	DPOINT	COUNTER_GET_CLEAR	R	No
Read digital input frequency value[1]	DPOINT	FREQUENCY	R	Yes

To do this	Use this DataType	Available Properties	Action	Scanned
Read the state of an off-latch	DPOINT	OFF_LATCH	R	Yes
Read and re-arm a high-speed off-latch associated with a standard digital input	DPOINT	OFF_LATCH_GET_CLEAR	R	No
Read the off-time duration of a digital input that has had an on-off-on transition[1]	DPOINT	OFF_PULSE	R	Yes
Read and clear the off-time duration of a digital input that has had an on-off-on transition[1]	DPOINT	OFF_PULSE_GET_RESTART	R	No
Read digital input total off time[1]	DPOINT	OFF_TIME	R	Yes
Read digital input total off time and restart[1]	DPOINT	OFF_TIME_GET_RESTART	R	No
Read the state of an on-latch	DPOINT	ON_LATCH	R	Yes
Read and re-arm a high-speed on-latch associated with a standard digital input	DPOINT	ON_LATCH_GET_CLEAR	R	No
Read the on-time duration of a digital input that has had an off-on-off transition[1]	DPOINT	ON_PULSE	R	Yes
Read and clear the on-time duration of a digital input that has had an off-on-off transition[1]	DPOINT	ON_PULSE_GET_RESTART	R	No
Read digital input total on time[1]	DPOINT	ON_TIME	R	Yes
Read digital input total on time and restart[1]	DPOINT	ON_TIME_GET_RESTART	R	No
Read the elapsed time from one off-on transition to the next off-on transition, or the elapsed time from one on-off transition to the next on-off transition of a digital input[1]	DPOINT	PERIOD	R	Yes
Read the elapsed time from one off-on transition to the next off-on transition, or the elapsed time from one on-off transition to the next on-off transition of a digital input[1]	DPOINT	PERIOD_GET_RESTART	R	No
Read the completion status of a period measurement[1]	DPOINT	PULSE_PERIOD_COMPLETE	R	Yes
Read a quadrature counter value[1]	DPOINT	QUAD_COUNTER	R	Yes
Read the current state (ON or OFF) of a digital point	DPOINT	STATE	R/W	Yes
Read or write the TPO on time percentage of an output point[1]	DPOINT	TPO_PERCENT	R/W	Yes
Read or write the proportional output (TPO) period of an output point[1]	DPOINT	TPO_PERIOD	R/W	Yes

[1] May require a firmware upgrade. Not available on legacy I/O units (UIO, EIO, SIO).

Examples

To do this	Use this
Read a counter of the digital input "My_Counter" with the counter feature enabled	[cont|ip|tcp:10.192.55.1:22001]DPOINT;COUNTER;My_Counter

ETHERNET PIDS

***CAUTION**: Be careful when using I/O Item Names through the PAC Control engine. Accessing I/O Item Names of remote serial I/O units is very slow and will likely cause severe performance issues. Also, systems with high volumes of Ethernet based I/O tags accessed through the controller may also have performance issues. In these cases, consider using "MMIO" tags to directly access the I/O data and circumvent accessing these values through the PAC Controller.*

For more information on how to achieve efficient I/O data throughput, see the Optimizing PAC Project System Performance Technical Note (form 1776).

Syntax

```
[Device|Protocol|Location]DataType;Property;VariableName
```

DataType and Property

Use PID as the DataType, and use a Property from the following table:

To do this	Use this DataType	Available Properties	Action	Scanned
Read the input value (also known as the process variable) of the PID	PID	INPUT	R	Yes
Read or write the output value of the PID	PID	OUTPUT	R/W	Yes
Read or write the setpoint value of the PID	PID	SETPOINT	R/W	Yes
Read or write the gain value from the PID	PID	GAIN	R/W	Yes
Read or write the integral value of the PID	PID	INTEGRAL	R/W	Yes
Read or write the derivative value of the PID	PID	DERIVATIVE	R/W	Yes
Change the Auto/Manual mode of the PID to Auto	PID	AUTO	R/W	Yes
Change the Auto/Manual mode of the PID to Manual	PID	MANUAL	R/W	Yes
Read or write the scan rate (update period) for a PID calculation	PID	SCAN_RATE	R/W	Yes

Examples

To do this	Use this
Read/Write the setpoint of Ethernet PID named MyEthernetPID	[CONT|ip|tcp:10.192.55.1:22001]PID;Setpoint;MyEthernetPID

MISTIC PIDS

***CAUTION**: Be careful when using I/O Item Names through the PAC Control engine. Accessing I/O Item Names of remote serial I/O units is very slow and will likely cause severe performance issues. Also, systems with high volumes of Ethernet based I/O tags accessed through the controller may also have performance issues. In these cases, consider using "MMIO" tags to directly access the I/O data and circumvent accessing these values through the PAC Controller.*

For more information on how to achieve efficient I/O data throughput, see the Optimizing PAC Project System Performance Technical Note (form 1776).

Syntax

```
[Device|Protocol|Location]DataType;Property;VariableName
```

DataType and Property

Use MPID as the DataType, and use a Property from the following table:

To do this	Use this DataType	Available Properties	Action	Scanned
Read the input value (also known as the process variable) of the PID	MPID	INPUT	R	Yes
Read or write the output value of the PID	MPID	OUTPUT	R/W	Yes

To do this	Use this DataType	Available Properties	Action	Scanned
Read or write the setpoint value of the PID	MPID	SETPOINT	R/W	Yes
Read or write the gain value of the PID	MPID	GAIN	R/W	Yes
Read or write the integral value of the PID	MPID	INTEGRAL	R/W	Yes
Read or write the derivative value of the PID	MPID	DERIVATIVE	R/W	Yes
Change the Setpoint Track Input in the manual mode parameter of the PID control word	MPID	SETPOINT_TRACK_INPUT	R/W	Yes
Change the Output Track Input in the manual mode parameter of the PID control word	MPID	OUTPUT_TRACK_INPUT	R/W	Yes
Enable or disable the PID to update its associated analog output channel	MPID	OUTPUT_ENABLE	R/W	Yes
Change the Auto/Manual mode of the PID to auto	MPID	AUTO	R/W	Yes
Change the Active/Reset mode	MPID	ACTIVE	R/W	Yes

Examples

To do this	Use this
Read/Write the gain of *mistic* PID named MyMisticPID	[CONT|ip|tcp:10.192.55.1:22001]MPID;Gain;MyMisticPID

MISTIC EVENTS/REACTIONS

CAUTION: *Be careful when using I/O Item Names through the PAC Control engine. Accessing I/O Item Names of remote serial I/O units is very slow and will likely cause severe performance issues. Also, systems with high volumes of Ethernet based I/O tags accessed through the controller may also have performance issues. In these cases, consider using "MMIO" tags to directly access the I/O data and circumvent accessing these values through the PAC Controller.*

For more information on how to achieve efficient I/O data throughput, see the Optimizing PAC Project System Performance Technical Note *(form 1776).*

Syntax

```
[Device|Protocol|Location]DataType;Property;VariableName
```

DataType and Property

Use ER as the DataType, and use a Property from the following table:

To do this	Use this DataType	Available Properties	Action	Scanned
Activate or deactivate a specific event/reaction	ER	SCAN_ENABLE	R/W	Yes
Determine if a specific event has occurred	ER	HAS_OCCURRED	R	Yes
Determine if the criteria for a specific event is currently true	ER	IS_OCCURRING	R	Yes

Example

To do this	Use this
Monitor if the event "My_Event" is occurring.	[cont|ip|tcp:10.192.55.1:22001]ER;IS_OCCURRING;My_Event

REDUNDANCY WITH THIRD-PARTY OPC CLIENTS

With PAC Display Professional and OptoDataLink, you can create redundancy in two ways:

- Redundant Ethernet links: when communication to the current network link is disrupted, they automatically switch to an alternate network link.
- Redundant Controllers:

The PAC Control User's Guide and the SNAP PAC System Specification Guide describe how these work, how they are different, and describe sample arrangement.

You can achieve a similar type of redundancy with third-party OPC client software by associating a secondary IP address to the primary IP address of each controller or I/O unit. The tags you use to configure this depend on whether you are connecting to a controller or an I/O unit, so you need to keep track of this information. It's important that you keep these IP addresses straight; you might want to use a table similar to the following to verify your work:

Controller or I/O Unit?	Primary IP Address	Secondary IP Address
Example: Controller	1.2.3.4	5.6.7.8

Here's what you do:

1. Configure both Ethernet ports of your *groov* EPIC processor or PAC controller. One will be *primary*, the other will be *secondary*.
   - For instructions on how to configure ports on *groov* EPIC processors, see "Configuring the Network Interface" in the *groov EPIC User's Guide* , form 2267. Remember to review the firewall settings before proceeding to the next step. Both ports need to allow communication to OptoMMP and PAC Controller.
   - For instructions on how to configure ports on PAC controllers, see "Configuring Devices" in the *PAC Manager User's Guide*, form 1704.
2. In the third-party OPC client:
   a. Create a group named **Config**. Make sure the group is configured to be *inactive*.
   b. Create and add the following items to the Config group.

For Controllers	For I/O Units	Specifications
[CONT|IP|tcp:1.2.3.4:22001]Scanner;FirstAltIp	[MMIO|IP|tcp:1.2.3.4:2001]Scanner;FirstAltIp	• Data type: String • Value: Write the IP address of the secondary Ethernet port (not the address of the primary Ethernet port).
[CONT|IP|tcp:1.2.3.4:22001]Scanner;IpActive	[MMIO|IP|tcp:1.2.3.4:2001]Scanner;IpActive	• Data type: Integer. • Value: Write a value of 1 (one) to this item.

For Controllers	For I/O Units	Specifications
[CONT|IP|tcp:1.2.3.4:22001]Scanner;IpEnableSet	[MMIO|IP|tcp:1.2.3.4:2001]Scanner;IpEnableSet	• Data type: Integer • Value: Write a value of 3 to this item. Note that this item is write-only.

For an explanation on the difference between the CONT and MMIO, see "Using MMIO and CONT Scanner Tags to Get I/O Data" on page 43.

Note that you can substitute FirstAltIp with AltIp; however, AltIp takes the data type String Array, which some third party OPC clients do not support. If you choose AltIp, these are the specifications:

- Data type: String array
- Number of indexes: 7 [0-6]
- Values: In index 0, write the IP address of the secondary port (not the address of the primary port).

Set the values of indexes 1 through 6 to EMPTY strings. Make sure none of them are NULL strings.

After you start the PAC Control strategy and OptoOPCServer, the system will switch to the secondary IP address if communication to the primary IP address fails.

Optional Commands

Four other optional commands are available. To configure them in your third-party OPC client, preface each one with **[CONT|IP|tcp:1.2.3.4:22001]Scanner;** (where 1.2.3.4 is the IP address of the primary Ethernet port of a controller) or **[MMIO|IP|tcp:1.2.3.4:2001]Scanner;** (where 1.2.3.4 is the IP address of the primary Ethernet port of an I/O unit).

- **IpEnableGet**

  Data type: Integer

  Gets the value of the Scanner;IpEnableSet item. A value of 3 means that automatic switching is enabled.
- **ConnectTimeout**

  Data type: Integer

  Sets the timeout value (in milliseconds) for the system to connect to the secondary IP address before beginning to sending data.
- **SendRecTimeout**

  Data type: Integer

  Sets the timeout value (in milliseconds) for transmitting data to and from the secondary IP address.
- **SendRecRetries**

  Data type: Integer

  Sets the number of retries for transmitting data to and from the secondary IP address.

5: Debugging and Troubleshooting

PROBLEMS DUE TO MICROSOFT FAST STARTUP/FAST REBOOT

Windows Fast Startup (called Fast Boot in older Windows) is a default Windows setting that helps computers boot up faster.

When Fast Startup/Fast Boot is enabled, Windows does not perform a fresh boot when you start your PC. Instead, it reuses some information about Windows that it saved when you shut down the PC.

Some applications (including PAC Display, OptoOPCServer, OptoDataLink, and SoftPAC) need a fresh boot up to properly function and communicate with other applications. When Fast Startup/Fast Boot circumvents this process, you may see issues such as these:

- Windows reports an error reading or finding a DLL file.
- Communication failures ("Cannot connect to scanner" errors) may prevent PAC applications from reading or updating tag values.

To resolve the issue, disable Windows Fast Startup/Fast Boot, and then restart the computer in order to get a fresh start.

Another solution that sometimes works is to uninstall PAC Project, reboot the computer, and then reinstall PAC Project.

For more information, see OptoKnowledgeBase article KB87235, Windows Fast Startup (Fast Boot) can cause issues.

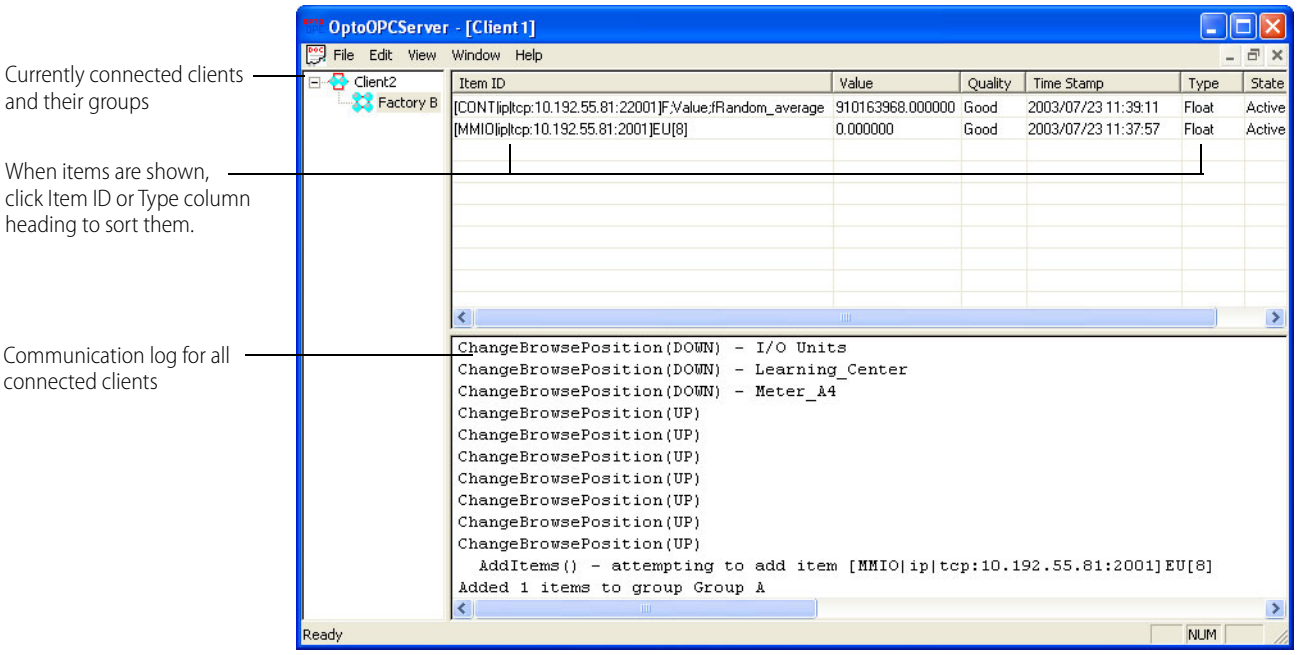
DEBUGGING IN THE OPTOOPCSERVER WINDOW

You may find data in the OptoOPCServer window useful for debugging. OptoOPCServer is normally launched by the client and runs invisibly; in order to see its data, you must manually start it *before* starting the client. You cannot open the OptoOPCServer window after the client has launched it.

1. Start OptoOPCServer:
   - In Windows, click the Windows Start button, and then click Opto 22 > OptoOPCServer 10.5.

   A blank OptoOPCServer window opens.
2. Launch either your OPC client or Prosys OPC Client.

   NOTE: To install and use Prosys OPC Client, see Appendix A: Using Prosys Client with OptoOPCServer.
3. In the Define the OPC Groups pane, click Add, and then create groups.
4. In the Define the OPC Items for the Group pane, add items as needed. For more information, see "Setting Up OptoOPCServer" on page 9.
5. Click inside the server window.



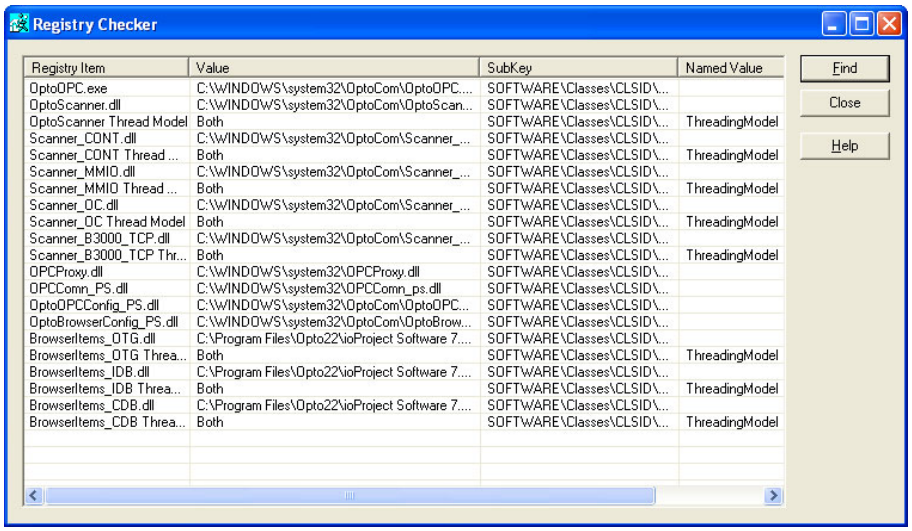
The server window shows all communication with currently connected clients. If you are monitoring more than one OPC client, each client will have its own set of windows. In addition, you can use the View menu to change the data you see:

- To see individual item IDs, choose View > Show Items in List. When items are not shown, the OPC server does not waste time updating the item list.
- To see item values change, choose View > Update Item Values.
- To change communication details shown in the window, choose View > Log Level.

USING THE REGISTRY CHECKER

OptoOPCServer includes the Registry Checker utility program, which you may be asked to run if you contact Opto 22 Product Support with a problem with OptoOPCServer. This program displays the registry values for each item listed in the file GUIDS.TXT, which shows how different software components installed with OptoOPCServer are registered on your computer. This information helps Opto 22 Product Support staff to diagnose possible problems with how OptoOPCServer software components are installed on your computer.

To open Registry Checker, choose Programs > Opto 22 > PAC Project > OptoOPCServer > Registry Checker.



TESTING A BROWSER DATABASE

If you need to check which browser databases are loaded on OptoOPCServer but your OPC client fails to connect to OptoOPCServer, you can use the Remove Browser Database function on the Opto Browser Configurator to view the list of browser databases. See "Removing a Browser Database from OptoOPCServer" on page 17. This will allow you to view which browser databases are loaded on that OptoOPCServer. You can also choose whether or not to remove a browser database. In addition, you can test a browser database using the Prosys OPC client available from Opto 22. For more information, see "F. Connect OPC Client to OptoOPCServer" on page 17.

DATA QUALITY MESSAGES

OptoOPCServer updates client applications with subscription data from configured devices. A value returned from an input or output point includes the value, data quality, and time stamp. This section explains the data quality messages, which may be reported in the client application as a text string, decimal, or hexadecimal value as shown in the following table.

Data Quality Description	Decimal	Hex
OPC_QUALITY_BAD	0	0x00
OPC_QUALITY_COMM_FAILURE	24	0x18
OPC_QUALITY_DEVICE_FAILURE	12	0x0C
OPC_QUALITY_GOOD	192	0xC0
OPC_QUALITY_LAST_KNOWN	20	0x14
OPC_QUALITY_NOT_CONNECTED	8	0x08
OPC_QUALITY_OUT_OF_SERVICE	28	0x1C
OPC_QUALITY_LAST_USEABLE	68	0x44

. OPC_QUALITY_BAD. (0 decimal; 00 hex) If previously the scanner was able to successfully scan the tag, and now it can't, the quality may be set to OPC_QUALITY_BAD. This occurs when a float value is a NAN (not a number). This can occur:

- With an analog value, when the analog module is not accessible to the brain for the rack it is mounted on. Check the voltage at that rack to make sure it is 5.00 to 5.20 VDC.
- When reading a floating point variable from the controller. A floating point variable is typically a NAN when:
  - A NAN analog point is moved to the floating point variable; or
  - There was a "divide by zero" error in the strategy logic.

OPC_QUALITY_COMM_FAILURE. (24 decimal; 18 hex) The scanner tries to connect to the appropriate device to obtain data for the tag. If it cannot connect, the quality is set to OPC_QUALITY_COMM_FAILURE. The scanner will keep trying to connect, but the quality will remain at OPC_QUALITY_COMM_FAILURE until it connects. OPC_QUALITY_ COMM_FAILURE could be caused by the following:

- A unit is not powered up.
- The Ethernet cable is not connected.
- The network is down.
- A hub/switch/router is offline.

OPC_QUALITY_DEVICE_FAILURE. (12 decimal; 0C hex) If the connection to the device is good, but either the device is busy or the tag is undefined at the device, the quality changes to OPC_QUALITY_DEVICE_FAILURE.

A device is busy when the controller's host ports have been *locked* temporarily, which may occur:

- During PAC Control strategy downloads
- While saving a strategy archive to the control engine (normally done in conjunction with a strategy download)
- While transferring files to the control engine using PAC Terminal
- While viewing control engine message queue messages via PAC Control or PAC Terminal

An *undefined tag* can be caused by a number of things:

- The strategy could have been modified and some of the tags may have been deleted or renamed, and then the modified strategy downloaded to the controller. Trying to access those tags will result in OPC_QUALITY_DEVICE_FAILURE because the tags are undefined.
- The tag name is misspelled. All tag names are case sensitive, so if the tag name in the strategy is in lowercase, for example, and the client is using the tag name with all caps, then it will not match.
- The strategy was modified and one or more of the tags were renamed.
- The strategy was cleared from the controller.
- You are trying to read a table element beyond the end of the table.

OPC_QUALITY_GOOD. (192 decimal; C0 hex) If the tag is scanned successfully, the quality changes to OPC_QUALITY_GOOD.

OPC_QUALITY_LAST_KNOWN. (20 decimal; 14 hex) If a tag has been successfully scanned and then communication to the device is lost, the quality changes to OPC_QUALITY_LAST_KNOWN.

OPC_QUALITY_NOT_CONNECTED. (8 decimal; 08 hex) When tags are added to the scanner, the quality is set to OPC_QUALITY_NOT_CONNECTED. Once the connection is established or fails, then the data quality is updated and will never return to OPC_QUALITY_NOT_CONNECTED for this connection. This initial attempt to connect may take a while, during which time the quality stays at OPC_QUALITY_NOT_CONNECTED. Every item quality is set to this when the item is first added, and there is no other use of this quality code.

OPC_QUALITY_OUT_OF_SERVICE. (28 decimal; 1C hex) This message occurs when a value is requested for an item that is not active or for an item whose group is not active.

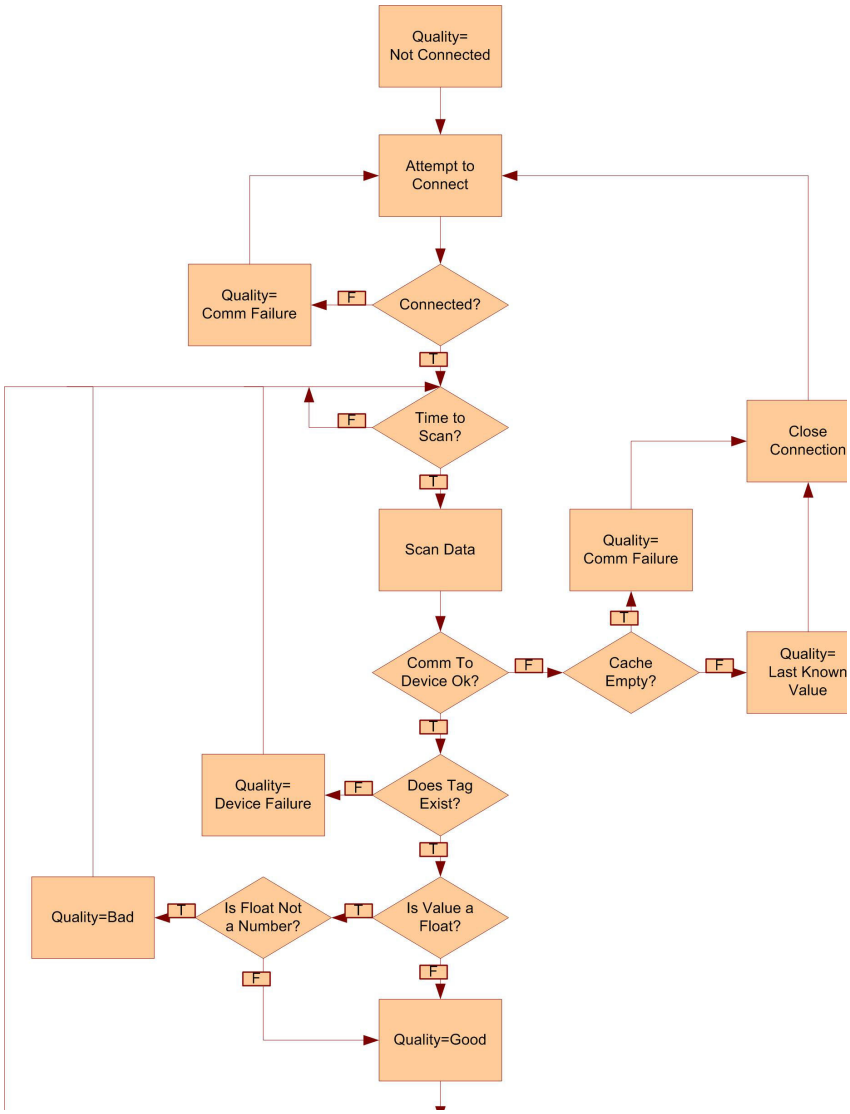
OPC_QUALITY_LAST_USEABLE. (68 decimal; 44 hex) This message occurs when the last value scanned is a NaN (not a number), which indicates that a floating point value is not valid. Typically, the value is related to an analog input point on an I/O unit in a situation where the brain could not communicate with the module.

The most common cause of this is low voltage at the rack. The voltage (as measured downstream of the fuse on the rack) should be 5.00 to 5.20 VDC. If there is any AC ripple or noise, then the fluctuations should stay within the 5.00 to 5.20 VDC range. If you are using an adjustable power supply, we recommend adjusting it so that the voltage (as measured downstream of the fuse on the rack) is about 5.15 VDC.

This error can also be caused by a divide by zero operation in strategy logic.

HOW DATA QUALITY IS DETERMINED

This simplified flowchart shows how the data quality is determined when OptoOPCServer updates the device subscription data. OptoOPCServer reports new data only when the value or the data quality has changed.



A: Using Prosys Client with OptoOPCServer

Prosys OPC Client is a free stand-alone OPC client application that you can use to test OptoOPCServer and the browser databases you create. This application is available as a free download from the Opto 22 FTP site.

In this appendix:

OBTAINING AND INSTALLING PROSYS OPC CLIENT

Follow these steps to download the Prosys OPC Client installer from the Opto 22 FTP site:

1. Open a browser and go to ftp://ftp.opto22.com/

   NOTE: FTP is not quite the same as normal web addresses, so if you click a link to ftp.opto22.com, your browser may put "http://" in front of the URL. If that happens, your browser will display an error message saying that it can't connect to the site. Instead, open your browser (or Windows File Explorer), type `ftp.opto22.com` *and then press the Enter key. If your browser still won't connect, type the full URL:* `ftp://ftp.opto22.com` *and then press Enter.*

2. Navigate to this location:

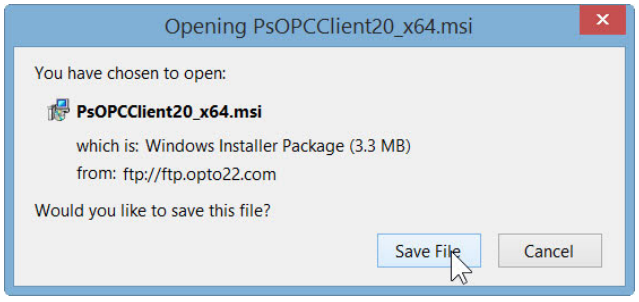
   `/Public_Folders_(Unsecured)/Archives_(Software_and_Firmware)/Software_Archives/ProSys_OPC_Test_Client/`

3. Click the installer link for your operating system:

   For a **32-bit** system, choose PsOPCClient20.msi.

   For a **64-bit** system, choose PsOPCClient20_x64.msi.

4. Click Save File to save the installer on your computer.



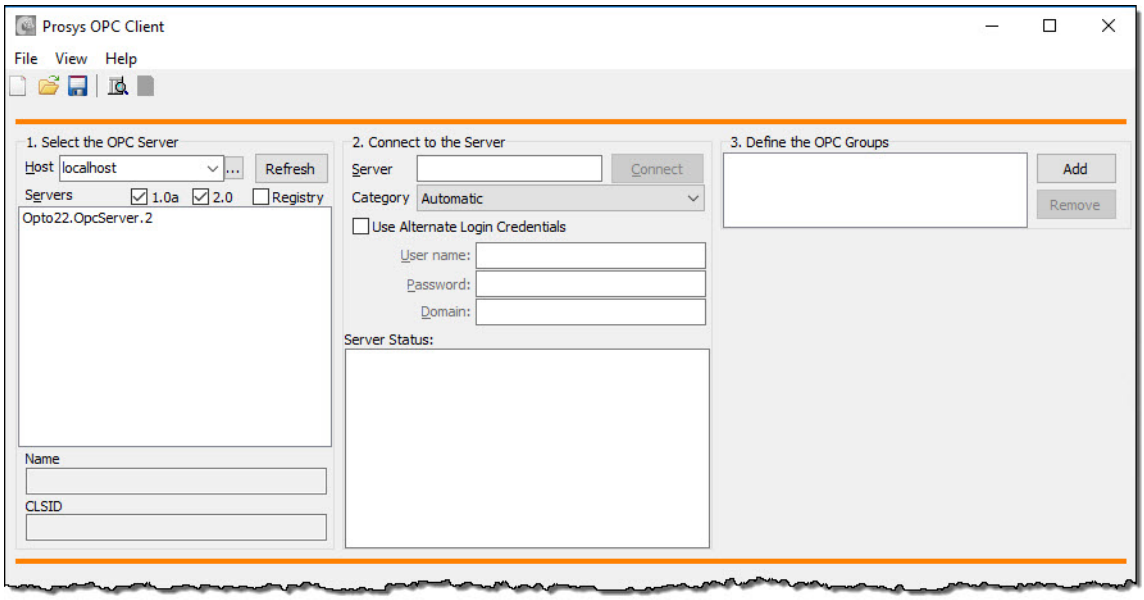
5. Click the downloaded installer file to start the Prosys OPC Client setup wizard. Follow the prompts to install the software.

CONFIGURING PROSYS OPC CLIENT

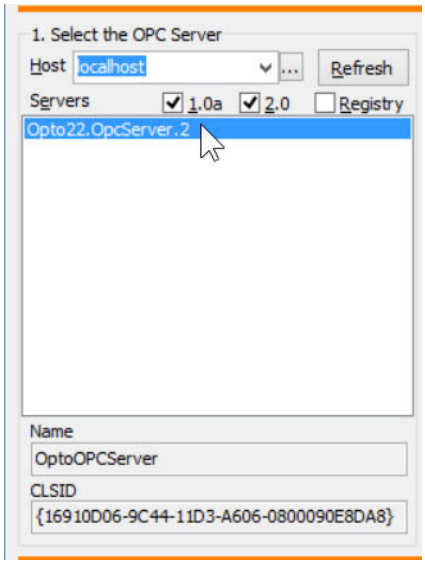
To run the client with OptoOPCServer:

1. Make sure the Prosys OPC Client is installed on your computer.
2. Open Prosys OPC Client:

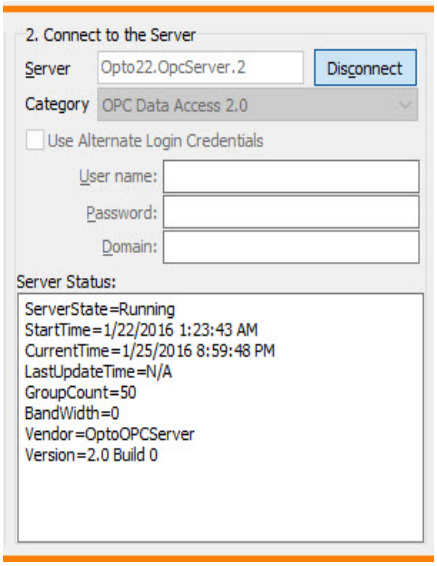
- In Windows, click the Windows Start button, and then click Prosys > Prosys OPC Client.



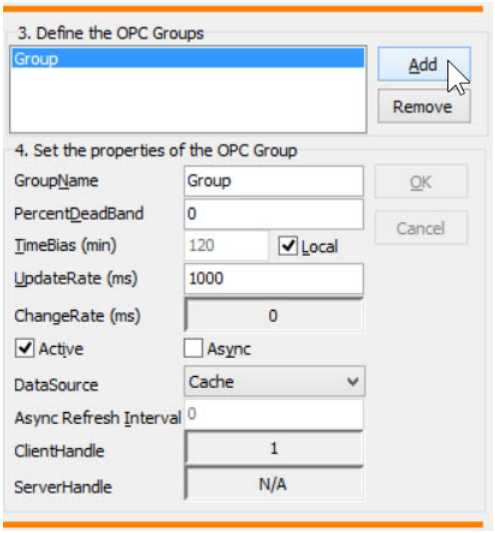
3. A list of registered servers on the local computer are displayed under "1. Select the OPC Server." The check box options allow you to narrow or expand the search for a specific type of server.
In the Servers list, select Opto22.OpcServer.2.



4. The Opto22.OpcServer.2 should now be listed as the Server in the Connect to the Server Pane. Click Connect to allow the OPC Client to connect to the server.



5. Next to 3. Define the OPC Groups, click Add.



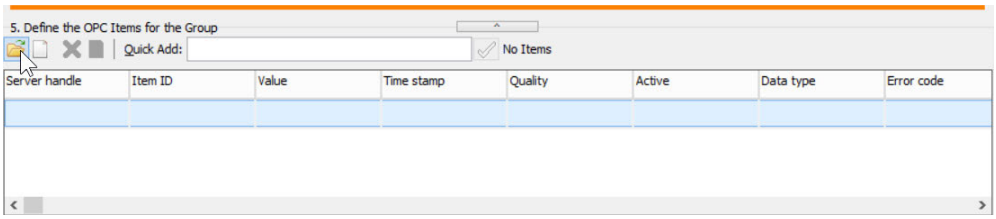
The client requires that you designate a "group" for the items the client is requesting. Groups are a way to separate items by the rate at which they need to be scanned. For example, temperature in a small chamber may need to be scanned more frequently, while an outdoor temperature may change much more slowly and need less frequent scanning. You can add as many groups as you need.

6. Enter a name in the GroupName field.
7. In the Update Rate (ms) field, enter the rate in milliseconds at which items in the group should be scanned.

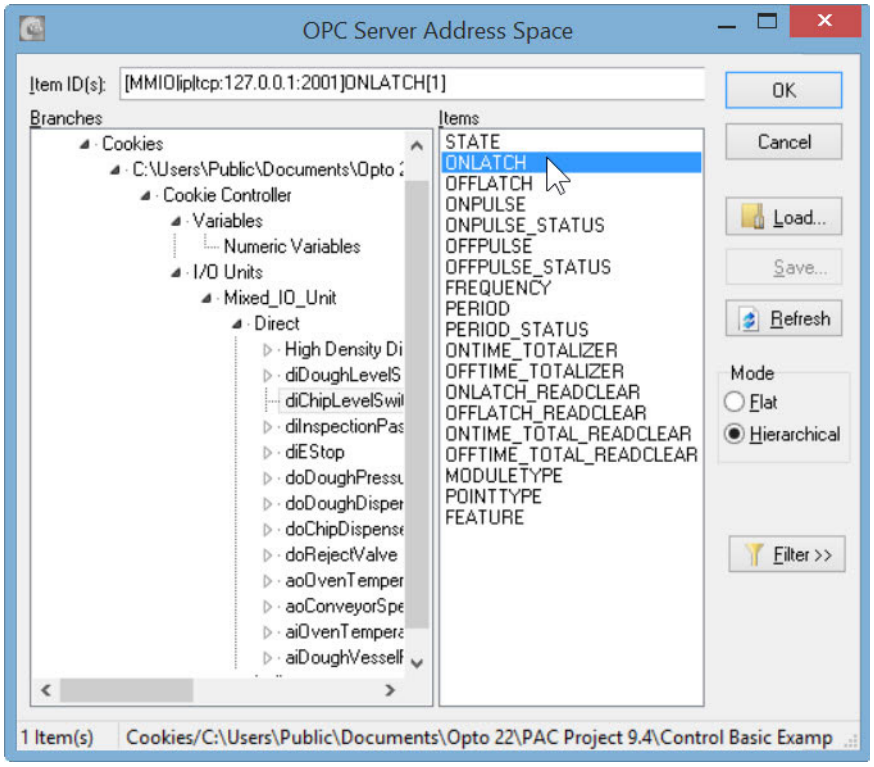
   The Active checkbox allows you to disable a group without having to remove it. The default communication is Synchronous, but can be changed to:

   – Asynchronous—by selecting the Async checkbox.
   – Subscription—by selecting the Async checkbox and setting the Async Refresh Interval to 0.

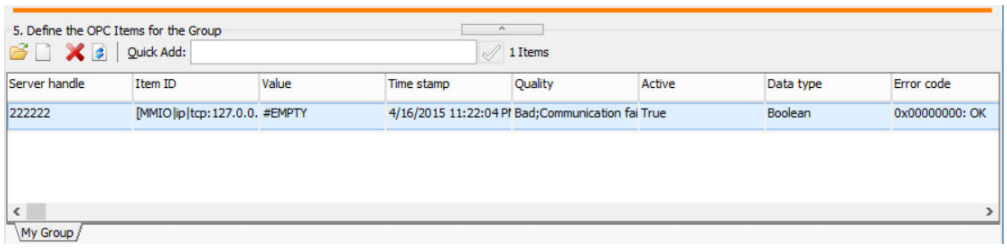
8. When you have finished defining the new group, click OK.
9. Under "5. Define the OPC Items for the Group," click the folder icon to open the OPC Server Address Space dialog box.



10. Expand the item branches to locate the tag you want to add.

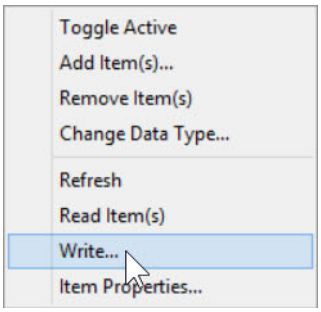


11. Highlight the tag, then click OK.
    The tag is added to your group.

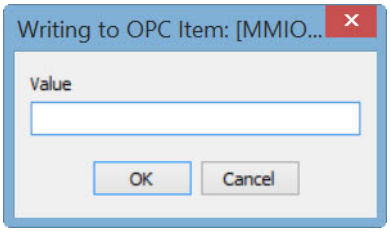


12. Repeat previous steps as necessary to add more tags to this group or to create another group of tags.

13. To write to a tag, right-click the tag name and choose Write from the pop-up menu.



14. In the Write to OPC item dialog box, type the value.



15. Click OK to send the new value to OptoOPCServer.

Index

U